1992年10月1日発行（毎月1回1日発行）第11巻10号通巻126号　昭和58年11月2日第三種郵便物認可

PERSONAL COMPUTER MAGAZINE for MZ, X1, and X68000

Oh!X

特集 DTMへの招待

Z-MUSIC用プリプロセッサ ZPP.X/LIVE in 5周年記念Special
新製品紹介 X68000用CD-ROMドライブ/SX-WINDOW開発キット
Small-C用横スクロールシューティングゲーム SLENDER HUL

10
1992

SOFT BANK オー！エックス
定価600円

SHARP

目の付けどころが、
シャープでしょ。

"感性"咲かせるワー

POWER WORKSTATION

インテリジェントなパフォーマンスを誇るX68000Compact XVIと多彩にラインアップされたペリフェラル。感性を刺激するクリエイティブなワークステーション環境が自在に構築できます。

- パーソナルワークステーション(2HD3.5インチFDDタイプ・本体＋キーボード＋マウス)
  CZ-674C-H(グレー) 標準価格**298,000**円(税別)
- 15型カラーディスプレイテレビ
  CZ-614D-TN(チタンブラック)・**-BK**(ブラック) 標準価格**135,000**円(税別)
  ■ディスプレイテレビ/CZ-6TU用RGBケーブル**CZ-6CR1** 標準価格**4,500**円(税別)
  ■ディスプレイテレビ/CZ-6TU用TVコントロールケーブル**CZ-6CT1** 標準価格**5,500**円(税別)
- 80MB内蔵用ハードディスクドライブ
  CZ-68HA 近日発売
- 5.25インチ増設用フロッピーディスクドライブ
  CZ-6FD5 標準価格**99,800**円(税別・接続ケーブル同梱)
- 光磁気ディスクユニット
  CZ-6MO1 標準価格**450,000**円(税別)
  ■SCSI変換ケーブル**CZ-6CS1** 標準価格**12,000**円(税別)
- 2MB増設RAMボード
  CZ-6BE2D 標準価格**54,800**円(税別・取り付け費別)
  ■2MB増設RAM**CZ-6BE2B** 標準価格**54,800**円(税別・取り付け費別)×2
  ■数値演算プロセッサ**CZ-6BP2** 標準価格**45,800**円(税別・取り付け費別)
- 48ドット熱転写カラー漢字プリンタ
  CZ-8PC5-BK(ブラック) 標準価格**96,800**円(税別)
- MIDIボード
  CZ-6BM1A 標準価格**26,800**円(税別)
- インテリジェントコントローラ
  CZ-8NJ2 標準価格**23,800**円(税別)

新登場

X68000 Compact用内蔵ハードディスクドライブパック

CZ-68HA…近日発売!!

資料請求券
X68000ペリフェラル
Oh!X
10係

クステーション環境。

X68000
PERSONAL WORKSTATION・XVI
Compact

GRAPHIC WORKSTATION

- パーソナルワークステーション(2HD3.5インチFDDタイプ・本体＋キーボード＋マウス)
  CZ-674C-H(グレー) 標準価格**298,000**円(税別)
- 21型カラーディスプレイ **CU-21HD** 標準価格**148,000**円(税別)
- 80MB内蔵用ハードディスクドライブ **CZ-68HA** 近日発売
- 光磁気ディスクユニット **CZ-6MO1** 標準価格**450,000**円(税別)
  ■SCSI変換ケーブル**CZ-6CS1** 標準価格**12,000**円(税別)
- 2MB増設RAMボード **CZ-6BE2D** 標準価格**54,800**円(税別・取り付け費別)
  ■2MB増設RAM**CZ-6BE2B** 標準価格**54,800**円(税別・取り付け費別)×2
  ■数値演算プロセッサ**CZ-6BP2** 標準価格**45,800**円(税別・取り付け費別)
- カラーイメージスキャナ
  CZ-8NS1 標準価格**188,000**円(税別)
  ■スキャナ用パラレルボード**CZ-6BN1** 標準価格**29,800**円(税別)
- カラーイメージジェット
  IO-735X-B(ブラック) 標準価格**248,000**円(税別)
  ■接続ケーブル**IO-73CX** 標準価格**5,500**円(税別)

STANDARD WORKSTATION

- パーソナルワークステーション
  (2HD3.5インチFDDタイプ・本体＋キーボード＋マウス)**CZ-674C-H**(グレー) 標準価格**298,000**円(税別)
- 14型カラーディスプレイ**CZ-608D-H**(グレー) 標準価格**94,800**円(税別)
- 5.25インチ増設用フロッピーディスクドライブ**CZ-6FD5** 標準価格**99,800**円(税別・接続ケーブル同梱)

TFT COLOR LCD WORKSTATION

- パーソナルワークステーション
  (2HD3.5インチFDDタイプ・本体＋キーボード＋マウス)**CZ-674C-H**(グレー) 標準価格**298,000**円(税別)
- 10.4型カラー液晶ディスプレイ**LC-10C1-H**(グレー) 標準価格**598,000**円(税別)
  ■接続ケーブル**AN-1515X** 標準価格**4,200**円(税別)

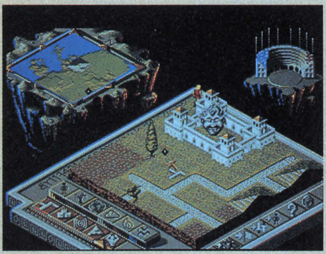
※カラー液晶ディスプレイを接続してご使用の場合、SX-WINDOW上のアプリケーション利用に限定されます。

- 大容量80MBのハードディスクドライブが、CZ-674Cのコンパクトな本体に内蔵可能。君のX68000 Compactがパワーアップ!
- 買ってすぐに使える、「SCSIドライバー＆フォーマットソフト」「取り付け用インターフェイス」同梱。

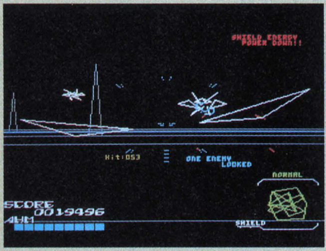
●お問い合わせは…
シャープ株式会社
電子機器事業本部システム機器営業部
〒545 大阪市阿倍野区長池町22番22号☎(06)621-1221(大代表)
電子機器事業本部AVCシステム事業推進室
〒162 東京都新宿区市谷八幡町8番地☎(03)3260-1161(大代表)

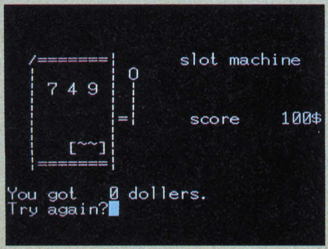
X 68000用CD-ROMドライブ

DōGA・CGアニメーション


ポピュラスII

リーディングカンパニー


AFTER REVIEW



（大）のショートプロぱーてぃ

Oh!X

C O N T

●特集

75 **DTMへの招待**

76 概論 MIDIをめぐる環境'92 中野修一

78 拡張された内蔵音源を使う Z-MUSICとPCM8 西川善司

80 Z-MUSICによる ローランド音源の基礎知識 瀧 康史

83 市販アプリケーションを使う MUSIC PRO-68K[MIDI]と仲よくなろう 高橋哲史

86 Z-MUSIC用MMLデータプリプロセッサ ZPP.X 岡本正和

98 おまけ Z-MUSICでバビンチョ（ゲーム応用編） 西川善司

●カラー紹介

65 Oh!X Graphic Gallery DōGA CGアニメーション講座

68 Oh!X reader's ぎゃらりぃ 暑中見舞いだ！

69 試用レポート X68000用CD-ROMドライブを使う 紀尾井誠

●THE SOFTOUCH

18 SOFTWARE INFORMATION 新作ソフトウェア/TOP10

20 TREND ANALYSIS

GAME REVIEW

22 ポピュラスII 丹 明彦

26 リーディングカンパニー 浦川博之

28 ネクタリス 影山裕昭

30 サークII 瀧 康史

32 AFTER REVIEW SIONII

〈スタッフ〉
●編集長／前田 徹 ●副編集長／植木章夫 ●編集／浅井研二 山田純二 豊浦史子 ●協力／有田隆也 中森 章 林 一樹 吉田幸一 華門真人 吉田賢司 影山裕昭 大和 哲 村田敏幸 丹 明彦 三沢和彦 長沢淳博 宮島 靖 金子俊一 浦川博之 石上達也 柴田 淳 御木徳高 ●カメラ／杉山和美 ●イラスト／永沢しげる 山田晴久 寺尾響子 ●アートディレクター／島村勝頼 ●レイアウト／元木昌子 ADGREEN ●校正／グループごじら

表紙絵：須藤　牧人

E　N　T　S

●シリーズ全機種共通システム

121 THE SENTINEL

122 実践Small-C講座(6) SLENDER HUL　柴田　淳

●読みもの

136 X-OVER NIGHT　第27話
パソコンドリームは終わった　高原秀己

●連載/紹介/講座/プログラム

34 概要紹介
SX-WINDOW開発キット　中野修一

36 (大)のショートプロぱーてぃ　その37
BATKEY.XとSLOT.BAS　大和　哲

38 DōGA CGアニメーション講座 ver.2.50（第2回）
打倒TORNADOへの第一歩（中編）　かまたゆたか

45 大人のためのX68000［第24回］
メモリ食い食いウラ画面　荻窪　圭

49 Oh!X LIVE SPECIAL
美少女戦士セーラームーン（X1・MIDIシーケンサ用）　蓮沼　勝
笑顔を探して（X68000・Z-MUSIC+ZPP用）　岡本正和
ドランゴンセイバー・エンディング（X68000・Z-MUSIC用）　西本英樹
Weapon・Shop（X68000・Z-MUSIC用）　矢部雅敏
TERROR FOR SALE（X68000・Z-MUSIC+PCM8用）　進藤慶到

60 ハードウェア工作入門〈28〉コンピュータアーキテクチャ編
加算器回路の工作実習　三沢和彦

66 響子 in CG わ〜るど［第17回］
キャラクタ　寺尾響子

111 よいこのSX-WINDOW講座（第10回）
スクロールバーに挑戦　中森　章

134 ANOTHER CG WORLD　寺尾響子

137 吾輩はX68000である［第17回］
マウスで簡易アニメーション　泉　大介

141 X68000マシン語プログラミング Chapter_24H
大きな数の話　村田敏幸

153 マシン語カクテル in Z80's Bar　第36回
お城と流れ星（その3）　金子俊一

157 Creative Computer Music入門（13）
ボーカルのかわりに……　瀧　康史

ペンギン情報コーナー……162
FILES Oh!X……164
Oh!X質問箱……166
STUDIO X……168
編集室から/DRIVE ON/ごめんなさいのコーナー/SHIFT BREAK/microOdyssey……172

1992 OCT.
10

UNIXはAT & T BELL LABORATORIESのOS名です。
Machはカーネギーメロン大学のOS名です。
CP/M, P-CPM, CP/Mplus, CP/M-86 CP/M-68K, CP/M-8000, DR-DOSはデジタルリサーチ
OS/2はIBM
MS-DOS, MS-OS/2, XENIX, MACRO80, MS C, MS-WindowsはMICROSOFT
MSX-DOSはアスキー
OS-9, OS-9/68000, OS-9000, MW CはMICROWARE
UCSD p-systemはカリフォルニア大学理事会
TURBO PASCAL, TURBO C, SIDEKICKはBOLAND INTERNATIONAL
LSI CはLSI JAPAN
HuBASICはハドソンソフト
の商標です。その他，プログラム名,CPUは一般に各メーカーの登録商標です。本文中では“TM”,“R”マークは明記していません。
本誌に掲載されたプログラムの著作権はプログラム作成者に保留されています。著作権上，PDSと明記されたもの以外，個人で使用するほかの無断複製は禁じられています。

■広告目次

アイビット電子……182(下)
アクセス……184
アプライド……180
エグザクト……11
計測技研……181
J & P……表3
シャープ……表2・表4・1・4-9
九十九電機……15
パソコンプラザオクト……178・179
ビクター音楽産業……10
P & A……12・13
マイクロウェアシステムズ……182(上)
マイコンショップ川口……177
満開製作所……176
ラインシステム……183(上)

SHARP

目の付けどころが、
シャープでしょ。

X68000 CompactXVI
NEWS

Opinion 1

(ハードディスクが使いたい。)

Compact専用の内蔵ハードディスクが登場します。SCSI仕様の80MB。場所を取らずに高速・大容量ファイル環境を実現します。

■内蔵用ハードディスクドライブ(CZ-674C専用)
CZ-68HA……………………………近日発売
※取りつけに関してはシャープお客様ご相談窓口にてご相談ください(取りつけ費別)。

さらに大容量をお望みの場合、外付け用のSCSI端子で一般のSCSIハードディスクも接続可能。フルピッチSCSI端子とハーフピッチSCSI端子を接続するためのSCSI変換ケーブルも用意しています。

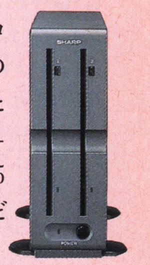
■SCSI変換ケーブル
CZ-6CS1………標準価格12,000円(税別)

CZ-6CS1

Opinion 2

(従来のソフト資産を活かしたい。)

これについても、Compact専用の外付け5インチフロッピーディスクユニットを用意していますから、従来の68シリーズの資産を有効活用できます。3.5インチと5インチの間でのデータのやりとりも可能。また、CZ-674C及びCZ-6FD5のスイッチ設定を変えれば、5インチソフトからの起動が可能になり、市販ソフトなどそのまま使えます。

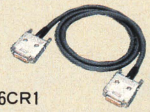


■増設用5インチ・フロッピーディスク・ユニット(CZ-674C専用)
CZ-6FD5……………………標準価格99,800円(税別)

Opinion 3

(ディスプレイテレビを接続したい。)

Compactは、従来のシリーズと比べ体積比44%と小さいため、コネクタの形状も異なっていますが、このケーブルを使用することにより、ディスプレイテレビやRGBシステムチューナーを利用できます。


CZ-6CR1

CZ-6CT1

■15型カラーディスプレイテレビ(スピーカー・チルトスタンド同梱)
CZ-614D-TN…………標準価格135,000円(税別)

■ディスプレイテレビ/CZ-6TU用RGBケーブル
CZ-6CR1……………………標準価格　4,500円(税別)

■ディスプレイテレビ/CZ-6TU用テレビコントロールケーブル
CZ-6CT1……………………標準価格　5,500円(税別)

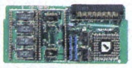
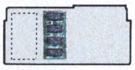
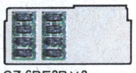
資料請求券
X68000ペリフェラル
Oh!X
10月

パーソナルワークステーション X68000 Compact XVIについてのご意見、ご要望にお応えします。

Opinion 4

(メモリ環境をパワーアップしたい。)

Compactは2MBのメインメモリを標準装備していますが、本体内で最大8MBまで拡張できます。

	容量	周辺機器
標準	2MB	——
拡張	4MB	 CZ-6BE2D
	6MB	 CZ-6BE2B
	8MB	 CZ-6BE2B×2

■2MB増設RAMボード CZ-6BE2D 標準価格54,800円(税別)

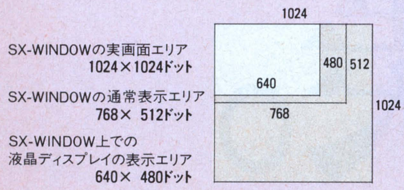
■2MB増設RAM CZ-6BE2B 標準価格54,800円(税別)

※取りつけに関してはシャープお客様ご相談窓口にてご相談ください(取りつけ費別)。

Opinion 5

(液晶ディスプレイとSX-WINDOWの関係は?)

液晶ディスプレイ(LC-10C1-H 標準価格598,000円・税別)の解像度は640×480ドット。Compactでは、従来のX68000シリーズの画面モードにこの画面モードをプラス。解像度の制約を受けないウィンドウ環境ならではの機能です。このようにSX-WINDOW環境の確立により、ハードウェアに依存しない快適な操作環境が実現します。



Opinion 6

(数値演算プロセッサはほんとに速い?)

ご存じのようにMPU68000自体は複雑な計算(浮動小数点演算)を単純な計算の組み合わせで行っています。X68000シリーズに装備されている浮動小数点演算パッケージ「FLOAT2.X」は、よく使う単純な組み合わせをまとめたもの。数値演算プロセッサは、いわばこのパッケージの機能を、ハードウェアで高速に実現し、MPUの負担を軽くするものです。アプリケーションプログラムの中には浮動小数点演算を必要としないものもあるため、すべてのプログラムが高速になるわけではありませんが、レイトレーシングなど大量の実数演算を必要とするソフトウェアの場合、飛躍的な実行速度の向上が期待できます。

■数値演算プロセッサ CZ-6BP2 標準価格45,800円(税別)

※数値演算プロセッサはCZ-6BE2D上に装着します。
※取りつけに関してはシャープお客様ご相談窓口にてご相談ください(取りつけ費別)。

X68000
PERSONAL WORKSTATION・XVI
Compact

本体+キーボード+マウス
2HD3.5インチFDDタイプ CZ-674C-H(グレー) 標準価格298,000円(税別)
14型カラーディスプレイ(ドットピッチ0.28mm)
CZ-608D-H(グレー) 標準価格94,800円(税別)

■お問い合わせは…電子機器事業本部システム機器営業部 〒545 大阪市阿倍野区長池町22番22号 ☎(06) 621-1221(大代表)
電子機器事業本部AVCシステム事業推進室 〒162 東京都新宿区市谷八幡町8番地 ☎(03) 3260-1161(大代表)

シャープ株式会社

SHARP

カラープリンタもスキャナも………

黒の統一美。

画像処理のベストマッチングシステム for X68000。

シャープ株式会社　〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)

入力から、編集、出力、複写まで、カラーの世界は、シャープです。

SHARP IS COLOR

▶INPUT

X68000用パラレルインタフェースを標準装備した高速コンパクト型イメージスキャナ。

カラーイメージスキャナ JX-220X……標準価格168,000円(税別)

●A4サイズの原稿を約50秒※1で高速読み取り●CCDセンサー採用。さらに中間調処理でシャープでリアルな画像を再現●ディザパターン指定機能※2や濃度補正機能※2など高度な画像処理機能で緻密な読み取りが可能●解像度200ドット/インチ(約7.9ドット/mm)。ズーム機能で1%きざみの拡大、縮小も可能●色ずれの少ない線順次(1走査)読み取り●X68000シリーズ用「スキャナツール」ソフトを標準装備●プリンタと直接接続することによりダイレクトプリント※3が可能●RS-232Cインタフェース/X68000シリーズ用専用パラレルインタフェースを標準装備。

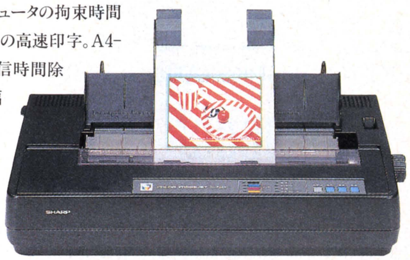
※1:A4、2値出力、コンピュータへの実転送時間。
※2:表記機能はJX-220X本体使用であり、付属ユーティリティ使用時は異なります。
※3:別売のパラレルインタフェースケーブル(JX-22PC標準価格12,000円(税別))が必要です。

▶OUTPUT

3種類の制御コマンドモードを搭載。質感も鮮やかに再現する高品位カラーイメージジェット。

カラーイメージジェット IO-735X-B……標準価格248,000円(税別)

●シャープ独自のIOシリーズコマンド(Gモード)に加え、NM-9900モード(Nモード)、ESC/P24-84C準拠モード(Pモード)をサポート。一般文書の作成から、各種デザイン、建築用パースなどのCAD分野に対応●発色性に優れた普通紙対応の新黒インキ採用。専用紙はもちろんオフィスでよく使われる普通紙にも鮮明カラー印字●プリントバッファメモリ(128KB)の内蔵で、ホストコンピュータの拘束時間を軽減●48ノズル(各色12ノズル)採用の高速印字。A4-1ページを※約90秒でプリント(データ受信時間除く)●ビジネス用途に適したB4横用紙幅対応●OHPフィルム(専用)にも鮮明プリント●ノンインパクト方式ならではの静粛印字●インキ補充は簡単、経済的なカートリッジ方式



※261×174mm領域

IO-735X-B 対応アプリケーション

●SX-WINDOW対応ペイントツール
Easypaint SX-68K
CZ-263GW 標準価格12,800円(税別)

●WYSIWYGを実現、ドローグラフィックソフト
CANVAS PRO-68K
CZ-249GS 標準価格29,800円(税別)

●オリジナリティを活かせるポップアップツール
NEW Printshop PRO-68K ver2.0
CZ-221HS 標準価格20,000円(税別)

●マルチワープロ PRO-68K
Multiword
CZ-225BS 標準価格32,000円(税別)

●高速カード型リレーショナルデータベース
CARD PRO-68K ver2.0
CZ-253BS 標準価格29,800円(税別)

●パソコン通信もできるメモリ常駐型ソフト
Teleportion PRO-68K
CZ-258BS 標準価格22,800円(税別)

●これからの高速通信をサポート
Communication PRO-68K ver2.0
CZ-257CS 標準価格19,800円(税別)

資料のご請求・お問い合わせはコンシューマーセンター／東日本相談室 〒162 東京都新宿区市谷八幡町8番地 ☎(03)3260-1161(大代表)・西日本相談室 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)・名古屋相談室 〒454 名古屋市中川区山王3丁目5番5号 ☎(052)332-2611(大代表)・福岡相談室 〒816 福岡市博多区井相田2丁目12番1号 ☎(092)575-2381

SHARP

目の付けどころが、
シャープでしょ。

待望のSX-WINDOW
開発支援ツール、登場。

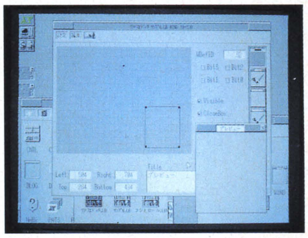
NEW

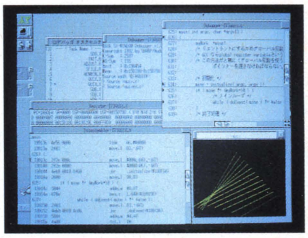
SX-68K

SX-WINDOW 開発キット

CZ-288LWD 9月発売予定

SX-WINDOW用のソフト開発に必要な開発ツールやサンプルプログラムを装備。プログラムの編集、リソースの作成、コンパイル、デバッグといった一連の作業をSX-WINDOW上で効率よく実行できます。初めてSX-WINDOW用のプログラムに挑戦する人にも、簡単に基本機能の理解ができる33種のサンプルプログラム付き。また各マネージャ解説と関数リファレンスの詳細なマニュアルも装備しています。





※本ソフトのご使用に際しては、メインメモリ4MB以上、SX-WINDOW ver2.0以上、C compiler PRO-68K ver 2.1以上が必要です。

キット構成

■開発ツール

●SXデバッガ
SX-WINDOW上で複数のプログラムを同時にデバッグすることができるソースコードデバッガ。

●リソースエディタ
SX-WINDOW上のリソースをリソースタイプごとの編集ウィンドウでビジュアルに作成・編集が可能。

●リソースリンカ
Cコンパイラやアセンブラで作成したリソースデータファイル(オブジェクトファイル)をリンクしてリソースファイルを作成。

●サンプルメイク
サンプルプログラムのコンパイル作業をSX-WINDOW上から、XC ver2のMAKE. Xを呼び出して、自動実行する簡易メイクユーティリティ。

■サンプルプログラム

●基礎編(23種)
各マネージャの基本的な機能のみを用いた基本動作の理解。

●応用編(4種)
基礎編での基本機能を応用した簡単なアプリケーションの作成。

●実用編(6種)
基礎/応用編での機能を駆使した、実用的なアプリケーションの作成。

■その他のファイル

●インクルードファイル
Cコンパイラとアセンブラ用の関数定義、データ定義ファイル。

●ライブラリファイル
Cコンパイラ用の関数ライブラリ。

マニュアル

●ユーザーズマニュアル ●プログラマーズマニュアル ●ファンクションリファレンス ●ライブラリリファレンス

X68000
PERSONAL WORKSTATION·XVI
Compact

本体＋キーボード＋マウス
2HD3.5インチFDDタイプ
CZ-674C-H(グレー) 標準価格298,000円(税別)

14型カラーディスプレイ(ドットピッチ0.28mm)
CZ-608D-H(グレー) 標準価格94,800円(税別)

●5.25インチ増設用
フロッピーディスクドライブ
CZ-6FD5
標準価格99,800円・税別
〔接続ケーブル同梱〕

●ディスプレイテレビ/CZ-6TU用RGBケーブル
CZ-6CR1 標準価格4,500円・税別

●ディスプレイテレビ/CZ-6TU用テレビコントロールケーブル
CZ-6CT1 標準価格5,500円・税別

●SCSI変換ケーブル CZ-6CS1 標準価格12,000円・税別

資料請求券
CZソフト
Oh!X
10係

開いてくださいウィンドウ、触れてくださいインテリジェンス。

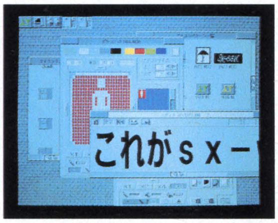
さらに拡がる、SXワールド。

●アウトラインフォント対応、ひらかれたウィンドウ環境。

SX-WINDOW ver2.0

CZ-287SS 標準価格12,800円(税別)

フォントマネージャを装備して待望のアウトラインフォントに対応。画面スクロール機能により、表示画面よりワイドなデスクトップ空間を駆使。アプリケーションのハンドリングに便利なシンボルトレイやアイコンメンテ、パターンエディタなど便利機能満載。



※SX-WINDOW ver1.0(CZ-259SS)およびSX-WINDOW ver1.1(CZ-278SS)をお持ちの方には有償バージョンアップを行います。

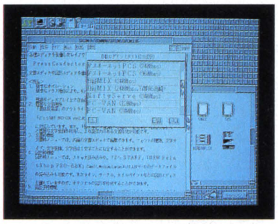
●マルチタスク機能をはじめ、通信環境がさらに充実。

Communication SX-68K

NEW

CZ-272CWD 9月発売予定

通信環境をさらに高めたウィンドウ対応の通信ソフトです。マルチタスク機能により他のアプリケーションソフトを実行中でも簡単に通信が可能。また、ホスト局をクリックするだけの自動ログイン機能、初心者にも簡単なプログラム機能、最新モデム(20種類)もフルサポートしています。



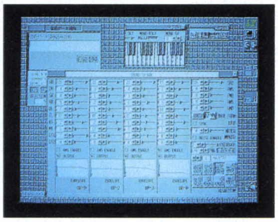
●多彩なサウンドクリエイトを実現するFM音源サウンドエディタ。

SOUND SX-68K

NEW

CZ-275MWD 標準価格15,800円(税別)

他のミュージックソフトで演奏中の音色を、簡単に作成・変更ができるマルチタスク機能、またエディット、イメージ、ウェーブの3つの編集/確認モードを装備。作成中の音色も50曲の自動演奏でリアルタイムに確認、編集できます。まさにミキサー感覚で音創りが楽しめるツールです。

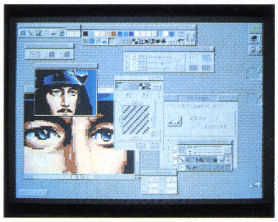


●ウィンドウ対応グラフィックツール。

Easypaint SX-68K

CZ-263GWD 標準価格12,800円(税別)

マウスによる簡単操作、65,536色中16色の多彩な表現、クリエイティブマインドに応えるウィンドウ対応ペイントツールです。同時に複数のウィンドウを開いて編集でき、各ウィンドウ間でのデータ交換もできます。



※SX-WINDOW対応ソフトの動作には、メインメモリ2MBおよびSX-WINDOW ver1.1以上が必要です。

充実のPROシリーズ

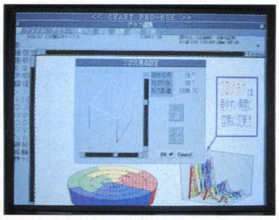
●ビジネスグラフチャート

CHART PRO-68K

NEW

CZ-267BSD 標準価格38,000円(税別)

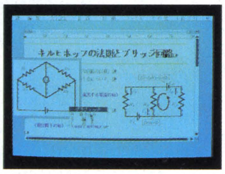
各種データベースで作成したデータをもとに、多彩なグラフが作成できます。3次元表示やグラフの複合機能も装備。データはMultiword,Press Conductor PRO-68Kに取り込むこともできます。



●グラフィック機能搭載の本格派ワープロ

Multiword ver 1.1

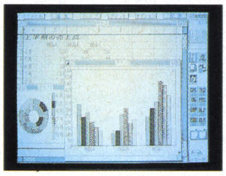
CZ-225BSD 標準価格32,000円(税別)



●簡単操作の統合型表計算ソフト

BUSINESS PRO-68K Popular

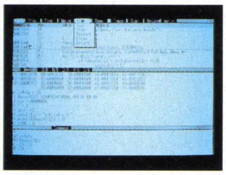
CZ-286BSD 標準価格28,000円(税別)



●各種ドライバ、ライブラリを追加

COMPILER ver2.1 PRO-68K

CZ-285LSD 標準価格44,800円(税別)

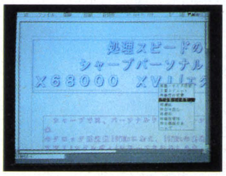


※有償バージョンアップ対応中。

●各種エディタ装備のレイアウトソフト

PressConductor PRO-68K

CZ-266BSD 標準価格28,000円(税別)



※以上のPROシリーズのソフトの動作にはメインメモリ2MB必要です。

※発売予定のソフトの画面写真は実物とは異なる場合があります。

●お問い合わせは…シャープ㈱電子機器事業本部AVCシステム事業推進室 〒162 東京都新宿区市谷八幡町8番地 ☎(03)3260-1161(大代表)へ。 シャープ株式会社

Victor

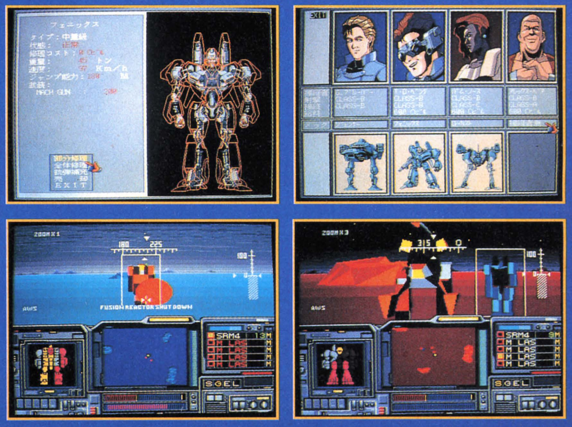
■米国生まれのそして米国で大評判を呼んだリアルタイム・シミュレーションゲームがオリジナルよりさらにバージョンアップして登場。
■「源 頼朝」か「源 義経」になって平家と戦いながら日本を手中におさめろ！
■変化に富んだ「馬追い」「忍者戦」などの4つのアクション・ゲームとリアルタイムの「合戦」がゲームを盛り上げる。
■刻々変化する情勢に的確に反応して軍勢を操りながらの平家を倒すか全国の城を制圧。

野望が渦巻く!! 貴族の世界か武士の時代か・・・
源氏と平家の壮絶な戦い。

©1992 VICTOR MUSICAL INDUSTRIES, INC.

■アメリカで爆発的なロボットブームを巻き起こした話題作いよいよ登場！
■3Dポリゴンの採用による迫真のバトル・アクション。
■プレイヤーが自らコックピットに乗り込みロボットを操縦、リアルなロボット・シミュレーションを体験。
■共に戦うクルーとして41人の傭兵から最大3人までの傭兵の採用により、戦略性もゲームの重要な要素。
■情報収集と賞金稼ぎによってグレイドの高いメックを手にいれて、150の惑星を舞台に任務を遂行。

奪われた聖杯を取り戻せ!
広大な宇宙空間に展開されるロボット・ウォーズ!



©1992 Activision. All rights reserved. Battletech is registered trademarks of FASA Corporation.
©1992 Victor Musical Industries, Inc.

発売元：ビクター音楽産業株式会社

通信販売 当社の商品をお近くのパソコンショップでお買い求めになれない場合、商品名、機種名、住所、氏名、電話番号を明記のうえ、下記住所まで定価プラス3%消費税分を現金書留にてお申し込み下さい。(送料無料) 〒151 東京都渋谷区千駄ヶ谷2-8-16 ビクター音楽産業㈱〈通信販売係〉

コミカル
ファンタジーアクション
ゲームRPG風
X68000
エトワールプリンセス
近日
発売予定
9,800円
〈税別〉
Étoile Princesse
みんなそろって怪物退治!!
●ダメージ制8方向
スクロールアクション
●MIDI対応!
●増RAM対応!
Étoile Princesse
T.K. Gulliverスーパー・アレンジで
おくる「アクアレス」全曲集だ!!
オリジナル・ロボットアクションゲーム
『アクアレス』のゲームミュージックCD
(アレンジ3曲含む全26曲)
好評発売中!!
APCG-4023 ㊙2,524円 ㊚2,600円
〈発売元〉
エグザクト
ExAcT
新潟市米山3-2-11 MKD.5ビル
☎(025)247-9160㈹
お求めは、全国パソコン専門店で。
取扱店がない場合は、住所・氏名・
電話番号と商品名を明記し、価格に
消費税を加算の上、当社宛に現金書
留にてお申し込み下さい。
※5インチ版のみの販売になります。
「ナイアス」「アクアレス」
3.5インチ版タケルにて好評発売中!!

P&Aならではの 新品パソコン 5年保証

《業界No.1の"P&Aメンテナンスサポート》

最高の保証システム

①業界最長の新品パソコン5年保証
（※モニター・プリンター３年間保証!! ※一部商品は除きます。）
②中古パソコンの１年間保証
（モニター・プリンター６ヶ月間保証）
③初期不良交換期間3ヶ月
（※新品商品に限らせていただきます）
④永久買取保証
⑤配達の指定OK!!
⑥夜間配送もOK!!
（※PM6:00～PM8:00の間 ※一部地域は除きます。）

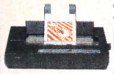
便利でお得な支払いシステム

①翌月一括払い手数料無料（ご利用下さい。）
②業界No.1の低金利
③月々の支払いは¥1,000より
④9ヶ月先からのスキップ払いOK!!
⑤84回までの分割、ボーナス併用OK!!
⑥カレッジクレジット
⑦ステップアップクレジット
⑧ボーナスだけで10回払いOK!!
⑨現金一括払いOK!!
（※商品・金額ご確認の上、銀行振込・現金書留にてご入金下さい。）

またまた

《増設メモリー&数値演算プロセッサ》計測技研

1 PRKII-02(2M)……定価¥ 55,000▶特価¥ 39,800
2 PRKII-04(4M)……定価¥ 90,000▶特価¥ 67,000
3 PRKII-06(6M)……定価¥125,000▶特価¥ 92,500
4 PRKII-08(8M)……定価¥160,000▶特価¥119,000
5 PRKII-12(2M)……定価¥ 85,000▶特価¥ 63,000
6 PRKII-14(4M)……定価¥120,000▶特価¥ 89,500
7 PRKII-16(6M)……定価¥155,000▶特価¥114,500
8 PRKII-18(8M)……定価¥190,000▶特価¥141,000
9 MC-68881RC……定価¥ 38,000▶特価¥ 27,000



カラーイメージジェット
■IO-735X-B
定価¥248,000
特価¥152,000
（送料・消費税込み¥157,590）

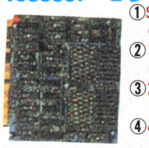
■Z's STAFF PRO 68K Ver.3.0
（ツアイト）（定価¥58,000）
特価 ¥36,500
（送料・消費税込み¥38,110）

■SX-68MII（MIDI）
（サコム）定価¥19,800
特価¥13,500
（送料・消費税込み¥14,420）

■HGS-68（スキャナ）
（HAL研）定価¥39,800
特価¥24,500
（送料・消費税¥26,265）

9/18～10/17

X68000メモリボード



①SH-6BE1-1M(600C専用)(I/Oデータ)定価¥25,000
（送料・消費税込み¥18,952）……特価¥17,900
②1MB増設RAMボード(ACE/PRO/PROII用)定価¥25,000
（送料・消費税込み¥16,583）……特価¥15,600
③2MB増設RAMボード(拡張スロット用)定価¥50,000
（送料・消費税込み¥32,239）……特価¥30,800
④4MB増設RAMボード(拡張スロット用)定価¥88,000
（送料・消費税込み¥55,620）……特価¥53,500

FDD(5インチ×2基)
■CZ-6FD5
（シャープ）（定価¥99,800）
P&A超特価!! TEL下さい。

注目!!冬のボーナス一括払い手数料(金利)無料（10月末/11月末/12月末のいずれかをご指定下さい。）

X68000 CompactXVI/XVI/XVI-HD

※送料¥2,000、消費税別

今月の特選!! 特価品

■Compact XVI さらにお安くなります。

●CZ-674C-H
●CZ-608D-H
●CZ-6FD5（5″FDD）
定価¥492,600
P&A超特価~~¥320,000~~
（※X68000サービスゲーム全て付いています。）
（モニターをCZ-606Dに変更の場合¥10,000を引いて下さい）

右記セットでお買い上げの方にもれなくプレゼント!!
①「ダウンタウン熱血物語（¥8,800）」
はもちろん、さらにその上、人気の
(イ)「ロードス島戦記（¥9,800）」
(ロ)「グラディウスII（¥9,800）」
(ハ)「ザ・プロサッカー68（¥9,800）」
(ニ)「信長の野望武将風雲録（¥9,800）」
(ホ)「ELLE（エル）（¥7,800）」
の中のいずれか2本をプレゼント!!

X68000-CompactXVI ●ディスケット10枚●ジョイカード2ケプレゼント中!! さらにお安くなります。

Ⓐセット：CZ-674C＋CZ-608D……定価¥392,800▶特価~~¥281,000~~

12回	24,700	24回	13,000	36回	9,000	48回	7,000	60回	5,900

X68000-XVI ▶セットでお買い上げの方に●ディスケット10枚●ジョイカード2ケプレゼント中!!

Ⓐセット：CZ-634C-TN＋CZ-606D-TN……定価¥447,800▶特価価格はTEL下さい。

12回	26,200	24回	13,800	36回	9,600	48回	7,500	60回	6,300

Ⓑセット：CZ-634C-TN＋CZ-614D-TN……定価¥503,000▶特価価格はTEL下さい。

12回	29,700	24回	15,700	36回	10,800	48回	8,200	60回	7,100

X68000-XVI-HD ▶セットでお買い上げの方に●ディスケット10枚●ジョイカード2ケプレゼント中!!

Ⓐセット：CZ-644C-TN＋CZ-606D-TN……定価¥597,800▶特価価格はTEL下さい。

12回	35,700	24回	18,800	36回	13,000	48回	10,200	60回	8,600

Ⓑセット：CZ-644C-TN＋CZ-614D-TN……定価¥653,000▶特価価格はTEL下さい。

12回	39,000	24回	20,600	36回	14,300	48回	11,200	60回	9,400

※上記のモニターを、CZ-606D（定価¥79,800）、CZ-604D（定価¥94,800）、CZ-607D（定価¥99,800）、CZ-605D（定価¥115,000）、CZ-608D（定価¥94,800）、CZ-614D（定価¥135,000）、CU-21HD（定価¥148,000）に変更の場合、TEL下さい。超特価で販売致します。

X68000シリーズ～P&Aスペシャルセット

（送料¥2,000・消費税別）

注目 スペシャルプレゼント!!
★ SUPER-HD には、
上記の①をプレゼント
★ PRO-II には、上記の
①＋(イ)～(ホ)の中の2本をプレゼント

ズバリ価格で大奉仕中

●ディスケット10枚、●ジョイカード2個プレゼント中

SUPER-HD P&A特選セット ★ハードディスク81MB搭載!!

Ⓐセット：■CZ-623C-TN（単品）…………定価¥498,000▶特価¥178,000
Ⓑセット：■CZ-623C-TN＋CZ-606D……定価¥577,800▶特価¥233,000
Ⓒセット：■CZ-623C-TN＋CZ-608D……定価¥592,800▶特価¥246,000
Ⓓセット：■CZ-623C-TN＋CZ-607D……定価¥597,800▶特価¥248,000
Ⓔセット：■CZ-623C-TN＋CZ-614D……定価¥633,000▶特価¥268,000
Ⓕセット：■CZ-623C-TN＋CU-21HD……定価¥646,000▶特価¥278,000

PRO-II P&A特選セット 限定

Ⓐセット：■CZ-653C（単品）……定価¥285,000▶特価¥138,000
Ⓑセット：■CZ-653C＋CZ-606D……定価¥364,800▶特価¥195,000
Ⓒセット：■CZ-653C＋CZ-604D……定価¥379,800▶特価¥197,000
Ⓓセット：■CZ-653C＋CZ-608D……定価¥379,800▶特価¥207,000
Ⓔセット：■CZ-653C＋CZ-607D……定価¥384,800▶特価¥209,000
Ⓕセット：■CZ-653C＋CZ-614D……定価¥420,000▶特価¥229,000
Ⓖセット：■CZ-653C＋CU-21HD……定価¥433,000▶特価¥239,000

X68000用ハードディスク（送料¥1,000 消費税別）

①LHD-FM100E（ロジテック）（外付）（¥99,800）
▶P&A超特価!!TEL下さい。
②LHD-FM200E（ロジテック）（外寸）（¥138,000）
▶P&A超特価!!TEL下さい。
③EFX-100B（エニックス）（¥118,000）
▶P&A超特価!!TEL下さい。
④EFX-140B（エニックス）（¥138,000）
▶P&A超特価!!TEL下さい。
⑤HD-J100（システムサコム）（¥128,000）
（送料・消費税込み¥87,550）▶特価¥84,000
⑥HD-J130（システムサコム）（¥148,000）
（送料・消費税込み¥104,030）▶特価¥100,000
⑦HD-J170（システムサコム）（¥189,000）
（送料・消費税込み¥121,540）▶特価¥117,000

プリンター（ケーブル用紙付）（送料¥1,000 消費税別）

■CZ-8PC5-BK（¥96,800）▶特価¥68,500
■CZ-8PK10（¥97,800）…▶特価¥71,000
■CZ-8PG2（¥160,000）▶特価TEL下さい。
■CZ-8PG1（¥130,000）▶特価TEL下さい。

モデム

■PV-M24B5
（AIWA）（定価¥39,800）
▶特価¥25,000
（送料・消費税込み¥26,780）

■MD-24FB5V
（オムロン）（定価¥39,800）
▶特価¥25,500
（送料・消費税込み¥27,295）

■FMMD-311G
（富士通）（定価¥35,800）
▶特価¥24,800
（送料・消費税込み¥26,574）

P&A特選パソコンラック（消費税別）（送料無料）

①3段¥7,900

1230(H) × 600(D) × 650(W)
消費税込み ¥8,137

②4段¥8,800

1230(H) × 600(D) × 650(W)
消費税込み ¥9,064

③5段¥12,500

1310(H) × 700(D) × 640(W)
消費税込み ¥12,875

全機種＝移動自由（キャスター付）●5段のみ＝キーボード収納可能、電源コード付(2.5m)(2P)

●本広告の掲載の商品の価格については、消費税は含まれておりません。 ●営業時間＝平日AM10:00～PM7:00、日祭AM10:00～PM6:00

アフターサービス万全
　全商品保証付。専門の担当がお客様の立場で対応します。
初期不良、輸送トラブルetc.
万が一初期不良、輸送トラブルが発生しました際には、即交換させていただきます。

★頭金なし!!
★即日発送!!

秋葉原でおなじみのP&AがズバリRX特価セールでご奉仕!!

秋葉原 で おなじみの P&A が ズバリ 超特価セール でご奉仕!!

- お近くの方は、お立寄下さい。専門係員が説明いたします。
- 本体単品でも受付します。詳しくは、お電話にてお問合せ下さい。
- ビジネスソフト定価の15%引きOK!! TEL下さい。
- 現金書留及び銀行振込でお申し込みの方は、上記商品の料金に3%加算の上でお申し込み下さい。詳しくは、お電話でお問い合せ下さい。

全国通販

X68000用ソフトコーナー （送料1ヶ～5ヶまで￥500・消費税別）

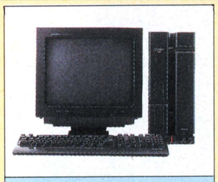
商品	定価	特価
◆Z's STAFF PRO68K Ver3.0(ツアイト)	定価￥58,000	特価￥36,500
◆Z's TRIPHONY デジタルクラフト(ツアイト)	定価￥39,800	特価￥27,000
◆テラッツォ(ハミングバード)	定価￥19,400	特価￥13,600
◆マジックパレット(ミュージカルプラン)	定価￥19,800	特価￥14,200
◆Gツール(ザインソフト)	定価￥28,000	特価￥18,600
◆たーみのる2(SPS)	定価￥17,800	特価￥13,000
◆Mu-1 Super	定価￥39,800	特価￥28,500
◆サイクロンEXPRESSα68	定価￥98,000	特価￥69,000
◆KAMIKAZE(サムシンググッド)	定価￥68,000	特価￥43,800
◆C-TRACE68 Ver3.0(キャスト)	定価￥98,000	特価￥68,500
◆G68K Ver.2 PRO	定価￥22,000	特価￥17,300
◆C&Professional Pack(マイクロウェアジャパン)	定価￥58,000	特価￥39,800
◆Final Ver3.2(エーエスピー)	定価￥38,000	特価￥29,000
◆CZ-213MSD MUSIC PRO68K	定価￥18,800	特価￥13,200
◆CZ-214MSD SOUND PRO68K	定価￥15,800	特価￥11,300
◆CZ-215MSD Sampling PRO68K	定価￥17,800	特価￥12,500
◆CZ-220BSD DATA PRO68K	定価￥58,000	特価￥40,000
◆CZ-224LSD The 福袋 Ver2.0	定価￥ 9,900	特価￥ 7,400
◆CZ-225BSD Multiword Ver1.1	定価￥32,000	特価￥23,000
◆CZ-243BSD CYBERNOTE PRO68K	定価￥19,800	特価￥15,000
◆CZ-247MSD MUSIC PRO68K[MIDI]	定価￥28,800	特価￥20,500
◆CZ-249GSD CANVAS PRO68K	定価￥29,800	特価￥22,000
◆CZ-251BSD Hyper word	定価￥39,800	特価￥29,400
◆CZ-253BSD CARD PRO68K Ver2.0	定価￥29,800	特価￥22,700
◆CZ-257CSD Communication PRO68K Ver2.0	定価￥19,800	特価￥15,300
◆CZ-258BSD Teleportion PRO68K	定価￥22,800	特価￥16,900
◆CZ-261MSD MUSIC studio PRO68K Ver2.0	定価￥28,800	特価￥21,200
◆CZ-263GWD Easypaint SX-68K	定価￥12,800	特価￥ 9,800
◆CZ-265HSD New Print Shop Ver2.0	定価￥20,000	特価￥15,400
◆CZ-266BSD Press Conductor PRO68K	定価￥28,800	特価￥22,000
◆CZ-284SSD OS-9/X68000 Ver2.4	定価￥35,800	特価￥25,600
◆CZ-285LSD C-Compiler PRO68K Ver2.1	定価￥44,800	特価￥32,500
◆CZ-286BSD BUSINESS PRO68K Popular	定価￥28,000	特価￥20,500
◆CZ-287SS SX-WINDOW Ver2.0	定価￥12,800	特価￥ 9,800

☆ゲームソフト25%OFF OK!!（一部ソフト除く）

周辺機器コーナー （送料￥500・消費税別）

商品	定価	特価
①CZ-8NS1	定価￥188,000	特価￥133,000
②CZ-6VT1	定価￥ 69,800	特価￥ 49,500
③CZ-6TU	定価￥ 33,100	特価￥ 23,900
④BF-68PRO	定価￥ 19,800	特価￥ 14,400
⑤CZ-8NM3	定価￥ 9,800	特価￥ 7,200
⑥CZ-8NT1	定価￥ 13,800	特価￥ 10,000
⑦CZ-6BE2A	定価￥ 59,800	特価￥ 42,800
⑧CZ-6BE2B	定価￥ 54,800	特価￥ 39,300
⑨CZ-6BE2D	定価￥ 54,800	特価￥ 39,300
⑩CZ-6BF1	定価￥ 49,800	特価￥ 35,800
⑪CZ-6BP1	定価￥ 79,800	特価￥ 57,000
⑫CZ-6BM1	定価￥ 26,800	特価￥ 19,300
⑬CZ-6EB1	定価￥ 88,000	特価￥ 63,000
⑭AN-S100	定価￥ 36,600	特価￥ 26,300
⑮CZ-6SD1	定価￥ 44,800	特価￥ 32,500
⑯CZ-6BN1	定価￥ 29,800	特価￥ 21,500
⑰CZ-6BV1	定価￥ 21,000	特価￥ 15,200
⑱CZ-6BC1	定価￥ 79,800	特価￥ 57,000
⑲CZ-6BG1	定価￥ 59,800	特価￥ 43,000
⑳CZ-6BU1	定価￥ 39,800	特価￥ 28,500
㉑CZ-6PV1	定価￥198,000	特価￥142,000
㉒CZ-6BS1	定価￥ 29,800	特価￥ 21,500
㉓CZ-8NJ2	定価￥ 23,800	特価￥ 17,500
㉔CZ-6BL2	定価￥298,000	特価￥214,000
㉕JX-100S	定価￥ 89,800	特価￥ 44,000
㉖JX-220X	定価￥168,000	特価￥121,000
㉗IO-735XB	定価￥248,000	特価￥152,000
㉘LC-10C1H	定価￥598,000	特価￥459,000
㉙CZ-6CS1(674C用)	定価￥ 12,000	特価￥ 8,900

P&A特選＝今月の中古特選品



●CZ-601C
●CZ-611D-TN
￥120,000

●CZ-634C-TN
●CZ-606D-TN
￥248,000

●CZ-644C-TN
●CZ-604D-TN
￥318,000

買取り価格

機種	価格	機種	価格
●CZ-634C	￥170,000	●CZ-602C	￥75,000
●CZ-644C	￥230,000	●CZ-612C	￥85,000
●CZ-604C	￥100,000	●CZ-652C	￥55,000
●CZ-623C	￥138,000	●CZ-662C	￥75,000
●CZ-603C	￥ 85,000	●CZ-611C	￥68,000
●CZ-613C	￥105,000	●CZ-601C	￥45,000
●CZ-653C	￥ 75,000	●CZ-600C	￥45,000
●CZ-663C	￥ 90,000		

下取り交換差額表

下取り ＼ 新品	CZ-634C モニターセット	CZ-644C モニターセット	モデル UX20セット	モデル CX20セット	9801DA2
CZ-623C モニターセット	150,000	270,000	70,000	160,000	130,000
CZ-613C モニターセット	190,000	290,000	100,000	190,000	160,000
CZ-652C モニターセット	230,000	340,000	150,000	240,000	220,000
CZ-604C モニターセット	180,000	290,000	100,000	190,000	160,000
CZ-600C モニターセット	230,000	340,000	150,000	240,000	220,000

中古・高価現金買取り 下取りOK!!

■まずはお電話下さい。
下取り専用 買取り電話 ▶ ☎03-3651-1884 FAX. 03-3651-0141
■下取り・買取りで、お急ぎの方は、直接当社に来店、または宅急便にてお送り下さい。

買取り価格…完動品・箱/マニュアル/付属品付の価格です。

- 下取りの場合……価格は常に変動していますので査定額をお電話で確認して下さい。（差額は、P&A超低金利クレジットをご利用下さい。）
- 買取りの場合……現品が着き次第、2日以内に買取り金額を連絡し、振込み、又は書留でお送り致します。
- 近郊の方は、P&A本店まで、直接お持ち下さい。即金にて、￥1,000,000までお支払い致します。

- 最新の在庫情報・価格はお電話にてお問い合せください。
- 買い取りのみ、または、中古品どうしの交換も致します。詳しくは電話にて、お問い合せ下さい。
- 価格は変動する場合もございますので、ご注文の際には必ず在庫をご確認下さい。
- 本商品の掲載の価格については、消費税は、含まれておりません。
- 現金書留及び銀行振込でお申し込みの方は、上記商品の料金に3%加算の上でお申し込み下さい。詳しくは、お電話でお問い合せ下さい。

《便利な超低金利クレジットをご利用下さい》

- 月々￥1,000円からOK!!
- ボーナス払いOK（夏冬10回までOK）
- 支払い回数 1回～84回
- お支払いは、8ヶ月先からでもOK!!

●定休日/毎週水曜日

通信販売お申し込みのご案内

〔現金一括でお申し込みの方〕
●商品名およびお客様の住所・氏名・電話番号をご記入の上、代金を当社まで、現金書留でお送りください。（プリンター・フロッピーの場合、本体使用機種名を明記のこと）

〔銀行振込でお申し込みの方〕
●銀行振込ご希望の方は必ずお振込みの前にお電話にてお客様のご住所・お名前・商品名等をお知らせください。
（電信扱いでお振込み下さい。）

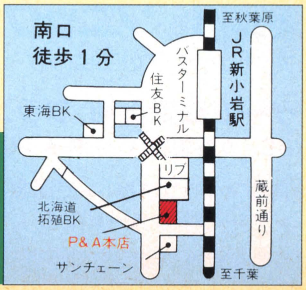
〔振込先〕さくら銀行　新小岩支店
普通預金　2408626　㈱ピー・アンド・エー

〔クレジットでお申し込みの方〕
●電話にてお申し込みください。クレジット申し込み用紙をお送りいたしますので、ご記入の上、当社までお送りください。
●現金特別価格でクレジットが利用できます。残金のみに金利がかかります。
●1回～84回払いまで出来ます。但し、1回のお支払い額は￥1000円以上。

超低金利クレジット率

回　数	3	6	10	12	15	24	36	48	60	72
手数料	3.0	4.0	5.5	5.5	8.5	11.5	16.0	21.0	27.0	33.0

注目!!冬のボーナス一括払い手数料(金利)無料（10月末/11月末/12月末のいずれかをご指定下さい。）



マイコン専門ショップ P&A
株式会社ピー・アンド・エー
〒124 東京都葛飾区新小岩2丁目1番地19号
☎03-3651-0148（代） FAX. 03-3651-0141
営業時間
平日：AM10:00～PM7:00
日祭：AM10:00～PM6:00

●価格は流通事情により変動致しますので、銀行振込・書留等の送付前に、あらかじめお電話にてご確認下さい。

The

スーパーファミコン

スーパーファミコンまるかじり！ 10/2号 9月18日㊎発売

定価380円（税込）
隔週金曜日発売

ドラゴンクエストV/エニックス

発売直前「ドラゴンクエストV」大特集！

特集：スーパーファミコン RPG全ガイド

スーパーファミコン版の全RPGをガイド。
それぞれのお買い得度を総チェック！

すぎやまこういちのゲーム漂流記
ゲスト エニックス：千田プロデューサー
「ドラゴンクエストV」はここが面白い！

攻略ガイド「三国志III」
「アクスレイ」「ガンフォース」
新作ガイド「ファイナルファンタジーV」
「ウィザードリィV」
「エルファリア」「ガンダムX」「天舞スピリッツ」ほか

BEEP! POWERFUL MEGA-MAGAZINE

MEGADRIVE

ビープ！ ▶メガドライブ◀

10月号 好評発売中
定価480円（税込）
毎月8日発売

特集
シューティングエクスタシー

メガドラ最新SHTの光と影、そして快感を総力特集！

続報！ ソニック2
アメリカ直輸入の「ソニック・ザ・ヘッジホッグ」の最新情報だ！

ゲーム誌史上初 フランス特派員大募集！
BEメガ特約雑誌『MEGA FORCE』が読者をフランスにご招待！

新作スクランブル ▶三国志III▶ぷよぷよ▶マジンサーガ▶ホリフィールド▶忍者外伝
▶Jリーグサッカー▶R. B. I. ベースボール▶パワーアスリート▶Cal. 60

別冊付録 ヴィクセン357 BASIC MANUAL
メサイヤの最新SLGを徹底攻略！

SOFT BANK
ソフトバンク出版事業部

最寄りの書店でお早めにお買い求めください

親切と安さの TSUKUMO

TSUKUMO 30TH YEAR SINCE 1962

新製品情報もツクモで
冬のボーナス先取り!

掲載商品２万円以上送料無料(一部地域を除く)

ツクモグローバルカード

大人気! 入会者募集中!

18才以上なら学生さんもOK!

国内・外で活躍!使って便利、持ってて安心!ツクモグローバルカードはジャックス・VISAとの提携カードです。ツクモ各店でのお買物がらくらくできる上に、国内はもとより海外での分割ショッピングもOK!
20才以上の方にはキャッシングカードも発行致します。
お申し込みは☎03(3251)9898又は店頭にて!

シャープX68000の事なら何でも揃うツクモにおまかせ!

秋葉原を歩き回る必要はありません。情報が沢山。分らない事は何でもお尋ね下さい。目に優しい10.4型カラー液晶ディスプレイ(LC-10CI)も取り扱い中!詳しくはお問い合わせ下さい。システムのご相談は☎03(3253)1899までどうぞ。

ワープロユースが中心で更にDOS/Vマシンのソフトを使う方へ

「書院パソコン」がお勧め

●スーパーアウトラインによる美しい印刷、すぐれた日本語処理能力●ワープロ「書院」の先進機能をそのまま継承●ハードディスク内蔵(Dタイプ)、OADG仕様、DOS/V対応●CPUは32ビット80386SXを搭載。

PC-WD1シリーズ 定価￥333,000より

ツクモ大特価展示販売中!

ホビーでガンガン使いこなす方へ

●X68000の未来を象徴するハイコンパクトなボディ(体積比44%) ●成熟するウィンドウ環境、使いやすさと高機能を追求したSX-WINDOW Ver2.0搭載 ●2HD 3.5インチフロッピーディスクドライブ2基搭載●カラー液晶ディスプレイ接続可能 ●X68000XVIの高性能を継承●VGAモードサポート(SX-WINDOWのみ対応)

X68000 Compact XVIセット

- CZ-674C-H……………￥298,000 (X68000 Compact本体)
- CZ-608D-H……………￥ 94,800 (0.28mmピッチCRT)
- 100MBハードディスク……サービス

合計定価￥392,800

ツクモ特価 **￥368,000**
(消費税別途￥11,040)
クレジット例(36回払・税込)
初回￥14,606＋月々￥12,600×35回

秋の夜長を大画面で楽しむ

あなたの部屋がミニシアター&迫力ゲームセンターに変身!

シャープ液晶ビジョンセット
XV-P1 定価￥220,000

今なら 「RGB信号→S端子変換ユニット」をサービス!

ツクモ特価 **￥198,000**

ツクモTSドライブ X68000用

目のつけどころがツクモでしょ。

X68000シリーズ専用3.5インチフロッピーディスクドライブ
TS-3XRシリーズ

〈仕様〉●3.5インチ2DD/2HD/2HCフォーマット対応の為、いろいろなフォーマットのメディアを読み書きができます。●ユーティリティソフト付属(ディバイスドライバー/フォーマッター)
※初代X68KはROM交換が必要です。

3.5インチ1ドライブ **TS-3XR1** 定価￥44,800
特価 **￥35,800**(消費税別途￥1,074)

3.5インチ2ドライブ **TS-3XR2** 定価￥57,800
特価 **￥46,800**(消費税別途￥1,404)

※只今開発中。X68000Compact XVI用外付け5インチFDD

コンピュータアート

スーパーグラフィックセット

慣れるとマウスがいらない。

WACOM製 ●SD-510C タブレット	￥98,000
●TJ-410A-2 接続ケーブル	￥6,000
●SP-200 スタイラスペン	￥10,000
サンワード ●Matier(マチエール)	￥39,800

合計定価￥153,800

ツクモ特価 **￥128,000**

● JX-220X A4サイズカラーイメージスキャナ……………定価~~￥168,000~~

X68000用ハードディスク

お勧めSCSIタイプハードディスク

100MB	ツクモ特価￥66,000
120MB	ツクモ特価￥76,000
200MB	ツクモ特価￥110,000

※SCSIボード(CZ-6BS1 定価￥29,800)は別売です。

X68000用MOディスク

ツクモはSONY MOディスクの正規代理店です。

これが今一番の人気者!
SONY 3.5インチ光磁気ディスクユニットセット

- RMO-S350(3.5"光磁気ディスクドライブ)…￥235,000
- SCSIケーブル……………￥6,900
- SCSIインターフェースボード……￥29,800

合計定価￥271,700

ツクモ特価販売中!

シャープ純正「CZ-6MO1」も特価販売中

X68000用メモリーボード

- ■1MB増設RAMボード(CZ-600C専用) ツクモ特価￥19,500
- ■1MB増設RAMボード(ACE/PRO/PRO2シリーズ用) ツクモ特価￥17,000
- ■2MB増設RAMボード(拡張スロット用) ツクモ特価￥33,800
- ■4MB増設RAMボード(拡張スロット用) ツクモ特価￥59,800

※計測技研のメモリーボードも取り扱っております。価格についてはお尋ね下さい。

パソコン通信

時代は9600ボーへ!!

- ■モデム 9600bps/MNP5 & CCITT V.42bis ツクモ特価￥69,800
- ■通信ソフト た〜みのる2 ツクモ特価￥14,000

電子文具

タイムマネージメントを管理する便利ツール

●従来の電子システム手帳用ICカードがそのまま使えます●次から次へと忙しい方の為の強力な助っ人●大画面・大容量・手書き入力で操作効率向上!

シャープ
電子マネジメント手帳
PV-F1 定価￥128,000

ツクモ特価販売中

更に、便利な名刺読み取り機「PV-BR1」(標準価格￥120,000)もお勧めです。

コンピュータミュージック X68000用

NEW Aセット

- SC-55……………￥69,000
- SX-68M-II……………￥19,800
- Mu-1 SUPER……………￥39,800

合計定価￥128,600
ツクモ特価 **￥99,000**
(消費税別途￥2,970)
クレジット例(18回払・税込)
初回￥6,596＋月々￥6,300×17回

NEW Bセット

- CM-300……………￥58,000
- SX-68M-II……………￥19,800
- Mu-1 SUPER……………￥39,800

合計定価￥117,600
ツクモ特価 **￥92,000**
(消費税別途￥2,760)
クレジット例(10回払・税込)
初回￥10,919＋月々￥10,000×9回

NEW Cセット

- CM-500……………￥115,000
- SX-68M-II……………￥19,800
- Mu-1 SUPER……………￥39,800

合計定価￥174,600
ツクモ特価 **￥141,000**
(消費税別途￥4,230)
クレジット例(15回払・税込)
初回￥11,300＋月々￥10,500×14回

※この他の組み合わせはお問い合せ下さい。☎03(3251)9911へ。

さらに拡がるSX-WINDOWワールド

●SX-WINDOW開発キット	CZ-288LWD	9月発売
●SX-WINDOW Ver2.0	CZ-287SS	~~￥12,800~~
●Communication SX-68K	CZ-272CWD	9月発売
●SOUND SX-68K	CZ-275MWD	9月発売
●Easypaint SX-68K	CZ-263GWD	~~￥12,800~~

冬のボーナス一括払(金利手数料なし)受付中!お問い合せ下さい。

通信販売のご注文は下記フリーダイヤルへ。

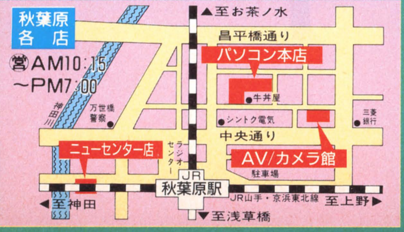
全国どこからでも通話料無料

受・注・専・用 フリーダイヤル **0120-377-999**

通販センター **03-3251-9911**

商品についてのお問い合わせは各店又は通販へ。

クレジット払い 月々￥3,000以上の均等払いも頭金なし、夏・冬ボーナス２回払いも受付中!	**各種リース払い** くわしくは各店にお問い合わせ下さい。ケースに合わせてご相談にのります!	**現金書留払い** 〒101-91 東京都千代田区神田郵便局私書箱135号 ツクモ通販センター Oh!X係
カード払い 通信販売での御利用カード、ツクモグローバルカード、VIPカード、セントラル、ジャックス※御本人様より電話で通販部へお申し込み下さい。	**全国代金引き換え配達** お申し込みは☎03-3251-9911へお電話１本! 配達日の指定もできます。	**銀行振込払い** 事前に☎でお届け先をご連絡下さい。三和銀行 秋葉原支店(普)1009939 ツクモデンキ



ツクモは「スーパーX PRO SHOP」です。

PRO STAFF ツクモ

パソコン本店 荒井

九十九電機株
〒101-91 東京都千代田区神田郵便局私書箱135号
★商品のご注文は在庫確認の上お願いします。★表示価格には消費税は含まれておりません。

ツクモパソコン本店2F
☎03-3253-1899(直通)

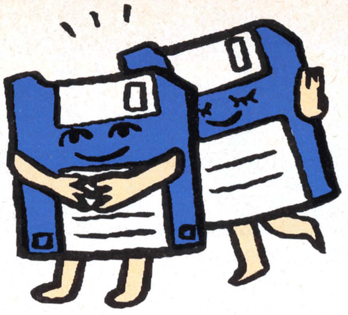
パソコン本店代表
☎03-3253-5599
(担当/荒井)
休毎週木曜

- ■ツクモニューセンター店 ☎03-3251-0987(担当/沢栄)休毎週木曜
- ■名古屋１号店 ☎052-263-1655(担当/吉高)休毎週火曜
- ■名古屋２号店 ☎052-251-3399(担当/横山)休毎週水曜
- ■ツクモ札幌店 ☎011-241-2299(担当/田口)休毎週木曜
- ■DEPOツクモ2番街店 ☎011-242-3199(担当/四条)休毎週木曜

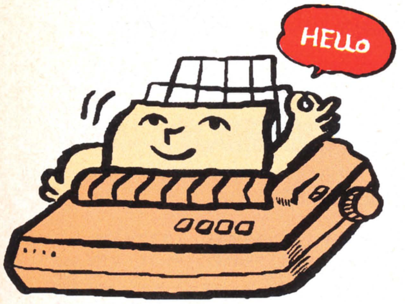
※定休日が祝日と重なる場合は営業致します。

話題のパソコン、ソフト、周辺機器を100ページ以上でレビュー

テストデータも交え、人がモノを語る



エンタテイメントも充実したビジュアルな誌面構成



“月刊PC”では投稿を募集します

新パソコン情報誌“月刊PC”は、パソコンユーザー参加型メディアです！あなたの生の声を、広く全国のパソコンユーザーに伝えます。

●READER'S REVIEW

実際にパソコン関連製品を使ってみて肌で感じた、製品の良かった点、がっかりした点を書いて、他のユーザーに教えてあげよう。

●秋葉原通信

東京・秋葉原や大阪・日本橋などパソコンのメッカでの出来事を知らせて下さい。

●ハード＆ソフトオリジナル利用法

マニュアルや活用本には書いていない、あなた独自のアイデア利用法を送って下さい。そのほか以下のテーマの投稿もお待ちしています。

●サポートの不平不満●読者の提案パソコン●長期稼働マシン自慢

レギュラー筆者に迫る破格の原稿料！

採用分の投稿には原稿料を支払います。投稿は原稿用紙でもフロッピー(DOSフォーマット)でもOK。ただし原稿は400字詰で上限7枚までに限らせていただきます。

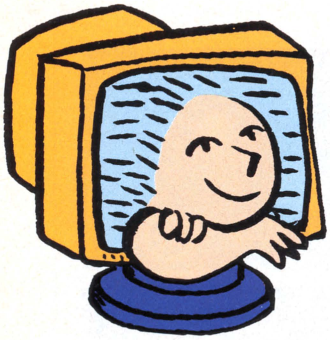
あ て 先 は

〒108 東京都港区高輪2-19-13 NS高輪ビル
ソフトバンク㈱／出版事業部
「月刊PC」編集部　投稿コーナー係

※月刊PCはお近くの書店でお買い求め・ご予約下さい

SOFT BANK
ソフトバンク株式会社
出版事業部　出版営業局
〒108 東京都港区高輪2-19-13 NS高輪ビル ☎03-5488-1360

購入可能なパソコン関連製品の情報をすべて網羅



10月17日創刊

月刊
PERSONAL COMPUTER
PC

いま大型のまったく新しいパソコン誌が誕生します。その名も「PC」。もちろんPersonal Computerの頭文字です。16ビットパソコンが誕生して10年。パーソナルコンピュータ市場は複数のアーキテクチャが勢力を競い合うなかで、量的拡大・多様化に加速がかかり、いよいよ混沌としてきました。もはやユーザーには、自分が使っていくパソコンの真の姿が見えにくくなってきています。いまどんなパソコンを選べば良いのか。パソコン関連製品を購入していくユーザーにとって、必要な製品情報を提供する“パーソナルコンピュータ総合情報誌”として、「ＰＣ」がデビューします。

［創刊号の主な記事］

■ユーザー投票により、パソコン、周辺機器、ソフトのそれぞれのベスト製品を選出
ＰＣが選ぶ、92年パソコン　ベストプロダクト！

■Windowsのための、ハイレゾ 486パソコンのベスト1！

■使いやすくて表現力豊かなワープロ王は？

■98、Mac、DOS/V、TOWNS、X68000
ゲーム上手で選んだベストなパソコンは？

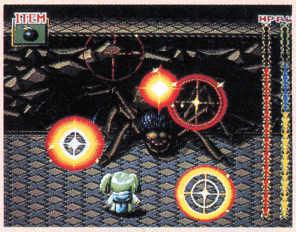
■特別付録
オリジナルディスクラベル

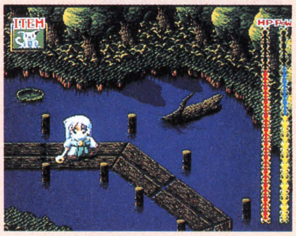
※誌面内容は予定ですので、変更になる場合があります

毎月18日発売/A4変型判/定価580円（税込）

SOFTWARE information

「シュートレンジ」そして「ポピュラスII」はすでに発売された。さて，発売日が遅れたり，決定しないソフトが続出するなか，ビクター音楽産業がここのところがんばっている。今度は「キャッスルズ」だ。



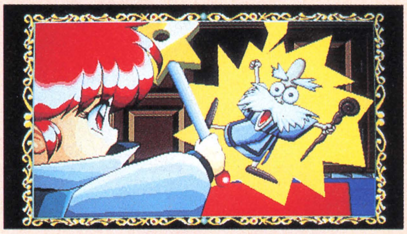


エトワールプリンセス

悪い魔法使いね～ちゃんを倒し，太陽の国の王子さまを助けることは前にも話したとおり。物語の舞台となるのは空に浮かぶフェイルランド（FAIL LAND）大陸で，ここにはそれぞれ特徴をもった星，月，太，地，水，火，風，金，銀，銅，竜，有羽，召喚の部族が住む地域が存在する（主人公のリルルは星の部族の王女）。

リリルが移動できる地域は各地域の解放（戦闘シーンクリア）や情報収集により増えていく。そして，全体マップモードでまだ解放していない地域を選んだ場合は，戦闘モードに入る。ここでは解放した種族の中から2人まで仲間を連れていくことができ，随時主役（Etoile）を変更することにより，それぞれに応じた攻撃が可能となる。ステージによってはある種族を組み入れないと攻略不可な面も存在する。

X68000用　5″2HD版　9,800円(税別)
エグザクト　☎025(247)9160



ファイナルファイトに変えただろう

1.	ファイナルファイト	1
2.	OVERTAKE	3↑
3.	グラディウスII	2↓
4.	スターウォーズ	8↑
5.	三國志III	5
6.	シムアース	9↑
7.	ポピュラスII	ー初
8.	出たな!! ツインビー	4↓
9.	ふしぎの海のナディア	7↓
10.	大戦略III'90	10

今月も王座を手にしたのはカプコンの「ファイナルファイト」。先月の発売以来，ハガキの推薦数も急上昇し，「OVERTAKE」の3倍以上という，TOP10としては異例の大差がつきました。格闘ゲームブームを引っ張っているカプコンの人気を改めて思い知らされます。2位のソフトが発売前，3位はピークを過ぎているということを考えると，有力なライバルはちょっと見当たりません。

その3位の「グラディウスII」ですが，実際のセールスのほうはまだまだ好調な模様。ハガキの推薦文にも満足度の高いものが多く，このゲームでMIDIの購入を決意するなど入れ込んでいる人もまだ数多くいます。“GAME OF THE YEAR”のときには「ファイナルファイト」と互角の戦いが期待できるのは間違いないですね。

「スターウォーズ」が再び持ち直してきましたが，これはいったい？“紅の豚を観ていたらやりたくなった”というコメントがあったけど，ホントにそれが原因？　夏休みに「スターツアーズ」でデートしたあとで，彼女に遊ばせる2段攻撃の需要があったという話もありますが，真相は闇の中。来月はどうなるのやら。

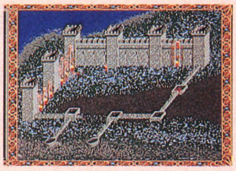
セールスが好調な「三國志III」ですが，推薦数のほうは伸び悩んでいます。まあ，“三國志フリーク”と読者ハガキを書いてくる人とは読者層が多少違うからしようがないのかも。ゲームをするのが忙しくてハガキを書いていられないという見方もありますが（ホント？）。

集計時点ではまだ発売前の「ポピュラスII」も票数を伸ばしてきました。前作の人気の再現となるかどうか注目したいところです。

では，来月まではすたらぴすた，べいびー。

（浦）

キャッスルズ



ビクター音楽産業の新作はまたまた海外移植モノ。この「キャッスルズ」では税率を調節しながら，各種職人を効率よく使いこなし，戦争に耐えうる城を建てていくのが目的である。舞台は中世のヨーロッパで，ドラゴンや魔法使いが登場する“Fantasy”モードと現実的な“Real”モードを選べる（メッセージが変わるだけだが)。難易度は4段階あり，好きなように城をつくれる“Peasant”から，収入や人口の少なさに制限され，戦争も激しい“King”まで。

X 68000用　5″2HD版　9,800円(税別)
ビクター音楽産業　☎03(3423)7901

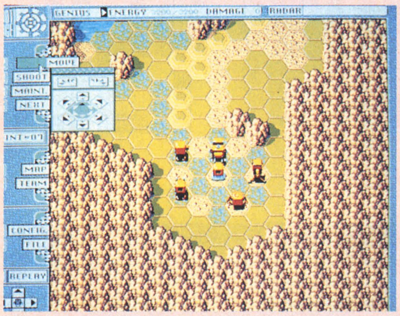
シュートレンジ

MSXのゲームでは，その独特のグラフィックセンスに定評のあったビッツーが，X68000にシミュレーションゲームで初参入。

ゲームの内容は6体のロボットを指揮して，敵のロボットをすべて破壊していくという近未来戦闘モノ。シンプルな仕上がりになっているが，特殊アイテムや索敵などシミュレーションゲームの要を押さえた出来となっている。グラフィックのセンスもやはり並ではない。

シナリオを勝ち上がるごとに，生き残ったキャラクターが継続して使用され，また，獲得した賞金によってアイテムによるパワーアップなどができる。このあたりはRPGとしての要素である。シミュレーションとしてはやや軽めなものの，シュートレンジ独特の世界はたっぷりと楽しめるのものとなっている。　(哲)

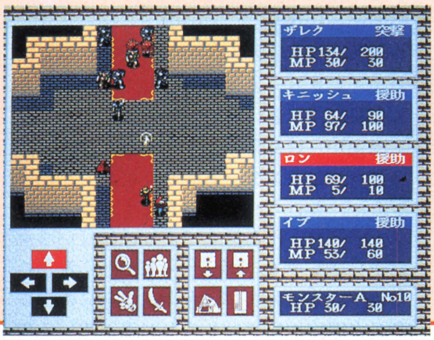
X 68000用　5″2HD版3枚組　9,800円(税別)
ビッツー　☎03(3479)4558



ヴェルスナーグ戦乱

失礼ながら，このゲームの発売は遅れに遅れている，といってしまえるのだが，開発は少しずつ進んでおり，なんとか近々発売できそうな感じだそうだ。今回届いたサンプルはダンジョン（どこかの建物？）の中を歩き回れるもので，戦闘の雰囲気もつかむことができた。味方が魔法を飛ばしたり，敵に近づいていって剣で斬りつける。敵もまたしかり。いわゆるタクティカルコンバットというやつであるが，基本的に戦闘は自動なのでラクである。

X 68000用　3.5/5″2HD版7枚組　9,800円(税別)
ファミリーソフト　☎03(3924)5727



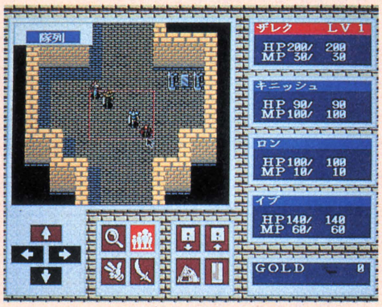
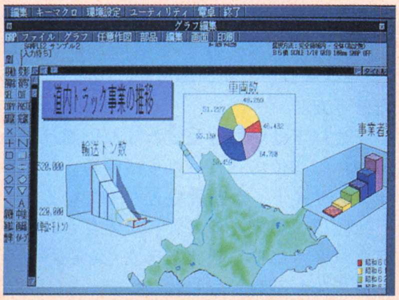
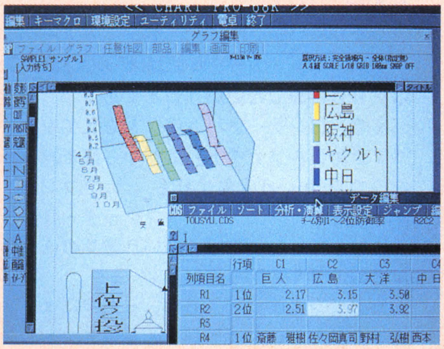


CHART PRO-68K





シャープから，グラフ作成用のビジネスツールが発売されることになった。

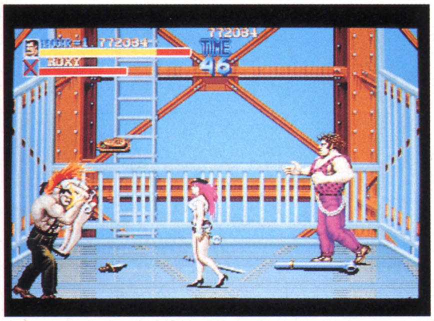
25種類のグラフに加え，ユーザーがグラフを自分でエディットしたり，それを保存することも可能。25種類ものグラフがあれば，一般的に使われるグラフをほとんどサポートしていると思われる。それらは2次元にとどまらず，3次元化された円グラフなどもある。また，個々のグラフデータはドロー系のデータで保存されているので，拡大/縮小/変形などが自由自在にできる。詳しくは次号にて。　(S.K.)

X 68000用　3.5/5″2HD版3枚組 38,000円(税別)
シャープ　☎03(3260)1161

TREND ANALYSIS

1992年7月の月間売り上げベスト10

POINT	タイトル	発売元	発売日
1220	ファイナルファイト	カプコン	'92/7/17
398	三國志III	光栄	'92/5/28
227	バトルテック	ビクター音楽産業	'92/7/10
138	スターウォーズ	ビクター音楽産業	'91/12/17
130	ジェノサイド2	ズーム	'91/12/8
105	グラディウスII	コナミ	'92/2/7
103	出たな!! ツインビー	コナミ	'91/12/6
48	ヨーロッパ戦線	光栄	'92/7/11
44	キャノンサイト	日コン連企画	'92/5
43	シムアース	イマジニア	'92/5/22



やはり「ファイナルファイト」がかなりのポイントを稼いで，トップになった。2位の「三國志III」と比べて，4倍ほどのポイントを獲得している。しかも，アンケートハガキを見ていると，売り切れで買えなかった人もいるようなので，数が揃っていればもう少しいったかもしれない。

このゲームは数年前に流行ったアーケードゲームからの移植である。登場キャラクターの動きや間合いを重視しており，プレイヤーには技，すなわちテクニカルな操作が要求される。現在隆盛を極めている格闘ゲームを確立させた作品といっていい。

X68000への移植では，細かいところに多少の違いはあるものの，全体的に見ればレベルの高い移植であり，ファンも納得がいく出来になっているようだ。

しかし，原作の名声とか，移植の素晴らしさよりも，もっともっと注目されている点がある。

つまり，いまいちばん人気があるアーケードゲームメーカー“カプコン”がX68000版を自ら発売したということである。

実際，アンケートハガキなどを見ても，“「ファイナルファイト」は昔から好きだったので”とかいう意見よりも，“カプコンだから”とか，“「ストリートファイターII」を移植してほしいから”という意見のほうが数の上でうわ回っている。

カプコンのほうも，そういった声には応える姿勢だというようなことをコメントしてはいるが，正式に「ストリートファイターII」を出すと発表したわけではない。

どうなるかは「ファイナルファイト」発売の手応えや，ユーザーの声，そして，カプコンのX68000市場に対する考え方に左右されるものと思われる。期待して待っていよう。

2位は「三國志III」。前回よりはポイントを大きく落としたものの，かなりのがんばりようである。さすがは光栄の主力シリーズの最新作，といったところだろうか。

光栄からは毎月のように作品，それもシミュレーションゲームばかりが発表されているが，飽きられることもなく，コンスタントな売り上げを見せているのにはいつもながら感心する。ユーザーも細かいところには不満をもちながらも，基本部分の出来のよさはほかにないものだと感じているのだろう。

3位の「バトルテック」は，ポリゴンでの3D戦闘シーンが目玉のロボットアクションゲーム。いつもたくさんのコメントをいただいている九十九電機の荒井さんは“パッケージデザインのおかげ”という見解だそうだ。

4位以下はいつもと変わりない面々であるが，8，9位の「ヨーロッパ戦線」と「キャノンサイト」が新登場となっている。

次回は「ポピュラスII」が注目されるが，月末（8月28日）発売と目立たないパッケージで少し不利かもしれない。

[データ集計協力店]（順不同）
九十九電機本店
J&P(渋谷/町田)
OAシステムプラザ横浜店
P&A
ラオックスGAME館

ウワサのソフトウェア（海外編）

AQUAVENTURA

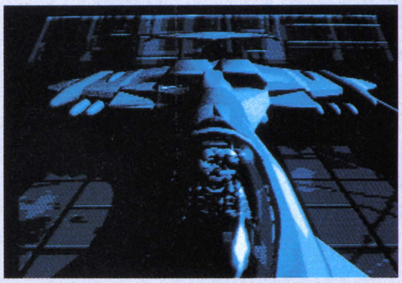
PSYGNOSISのゲームはデモが有名で，FM TOWNSの「フラクタルエンジン」にも多数のデモが収録されている。このことはもう皆さんご存じだろう。

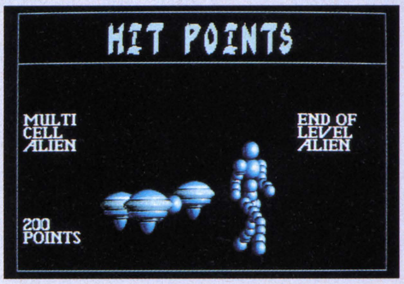
この「AQUAVENTURA」もその例にもれず，1枚がオープニングデモ，もう1枚がゲームディスクとなっていて，伝統は無事守られている。デモはいつもながらの出来で（おそらく自社開発のレンダリングシステムで制作されている）CGアニメーションには目を奪われる。ちょっと動きが人形アニメっぽいが，それもいかにもという感じでよろしい。

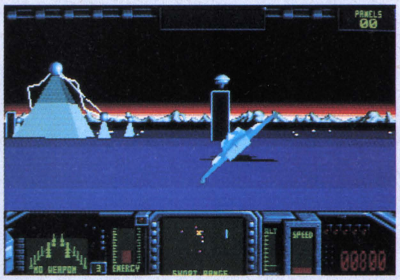
しかし，伝統を踏襲しているのは外見だけであって，実は偉大なる進化を遂げていた。

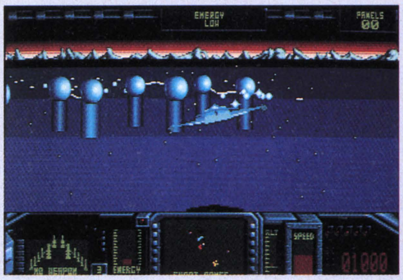
な，な，なんとゲームディスクからもブートできるのである。いままでデモを見なければゲームを遊ばせてくれないのが決まりだったのに。“うーん。PSYGNOSISはちゃんとデモ以外のところにも気を配るようになったのか”。当たり前の変化ではあるが，そういう感想をもたせるようなメーカーなのだ。

さて，内容はといえば，3Dゲームである。ポリゴンで描かれたキャラクターと，スプライト（純粋にいうとスプライトではないが）の拡大/縮小で表現されたキャラクターがうまく混在していて，独特の雰囲気をもっている。自分の戦闘機や，ターゲットとなる水上建築物はポリゴンで，空飛ぶ円盤状の敵や，ヘビやムカデのようなボスキャラはスプライト（レイトレの球体っぽいのが多い）という具合である。









ゲームの目的も単純で，敵が出てくるトビラを壊しながら，太陽エネルギー吸収装置を壊す。すると，ピラミッドのバリアが解除されて破壊できる。破壊すると，そこからボスが出てくるのでやっつける。やっつけると敵がいっぱいいるトンネルに入れられて，次の面に向かう。その繰り返しである。難易度もそんなには高くないのでわりと楽しめる。

面と面の間には，もちろん3Dポリゴンによるアニメーションなどもあるが，それよりもゲーム開始前の敵キャラ得点表示と，バラバラになったはずの自機が海中で発見されるゲームオーバー画面が気に入った。

発売元　PSYGNOSIS

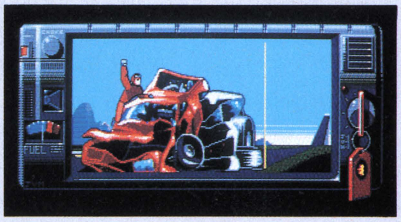
ウワサのソフトウェア（海外編）

RACE DRIVIN'

3Dポリゴン表示によるドライビングゲーム，「HARD DRIVIN'」。このアーケードゲームは日本でも有名だったし，メガドライブ版というのもあった。

コースも，コースわきにある建物も，自分の車も，ほかの車も（ライバルや対向車），すべてがポリゴンで描かれていることも特徴のひとつであるが，それ以上にコースのユニークさが際立っている。

コースはひとつであるが，スタート直後にスピードコースとスタントコースに分岐している。スピードコースはアップダウンこそあるものの，まあ普通の道路。しかし，スタントコースはその名のとおり，アクロバット走行の嵐。勢いをつけてジャンプしないと落ちてしまう橋，空中でグルリと回るループ，スピードが乗っていないとずり落ちてしまうバンクと，それまでにない ジェットコースター感覚の仕掛けが散りばめられていた。



このゲームはAMIGAなどの海外パソコンでもずいぶん昔に発売されていて，コースのコンストラクション機能がついた「HARD DRIVIN' II」というのも出ていた。

そして，「RACE DRIVIN'」である。まずは，仕様上の主な変更点を記述していこう。

車は6つ。“SPEEDSTER”，“SPORTSTER”“ROADSTER”の3種類のそれぞれにマニュアル車とオートマチック車がある。

コースは3つ。「HARD DRIVIN'」と同じ“SPEED & STUNT”，そして“SPEED RACING, AUTOCROSS”“SUPER STUNT TRACK”である。コース上の仕掛けには，“JUMP LOOP（UPSIDE DOWN）”や“FULL PIPE”“CORKSCREW LOOP”などが加わった。また，背景が取り込み画像になったのはいいが，少しミスマッチになっている。通信ケーブルでコンピュータどうしをつないで，対戦することも可能になっている。

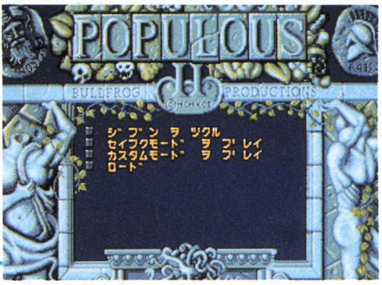
で，結局面白くなったのかというと，実はそうでもない。操作性は前作よりはるかに悪くなっているし（入力に対するレスポンスが悪い），画面のバグリ（ポリゴンの計算に失敗するのか，わけのわからない画面になってしまう）もさらに多くなっている。ポリゴンをあれだけ動かすには計算上の手抜きをするのは当たり前だが，それにしても多すぎる。

発売元　DOMARK

神を目指して放火三昧

Tan Akihiko
丹　明彦

2年前，パソコンゲーム界に大旋風を巻き起こした「ポピュラス」が，いちだんとパワーアップして帰ってきた。前作の面白さはそのままに，視覚および音響効果を向上させ，強烈でリアルなゲーム世界を作り上げている。



ポピュラスとは

「ポピュラス」は神と悪魔の戦争をシミュレートしたゲームである。神と悪魔はそれぞれを信奉する種族を従えて戦い，相手の民族を全滅させたほうが勝つ。通常はプレイヤーが神を，コンピュータが悪魔を受けもつが，通信ケーブルで2台のコンピュータを接続することにより，人間どうしの対戦も楽しめるようになっている。

民衆はプレイヤーである神の意思などおかまいなしに歩き回り，平地を見つけては家を建てる。神は土地を造成して自分の民の繁栄を援助したり，地震などの災害を起こして敵の民の繁栄を妨害したりすることができる。

神と悪魔の双方の民衆にはリーダーと呼ばれる代表がいる。神はリーダーを通じて民衆を導くことができる。また，リーダーを騎士と呼ばれる殺戮マシンに昇格させ，敵の民衆を蹂躙させることもできる。

民衆が神を信仰すると，マナと呼ばれるエネルギーが蓄積される。このマナの量は神の起こせる奇蹟（土地の造成や災害）に影響するので，上手に土地を造成し，民衆を導き，人口を効率よく増やすことがゲームに勝利する基本となる。逆に，敵の土地に災害を起こすと，敵の民衆の人口を減らし，敵が災害復旧のためにマナを消費するので結果的に自分に有利に働く。これも基本である。

X68000用　5″2HD版2枚組　12,800円（税別）
イマジニア　☎03(3343)8911

ゲームはリアルタイムで進行する。自分の民を繁栄させつつ相手の民を攻撃するために，的確な判断と迅速な行動が要求される。ときには善き神であることを放棄し，あくまで冷酷に作戦を展開していくことも必要になる。

「ポピュラス」は，コンピュータ相手に遊ぶのも面白いのだが，対戦ポピュラスの面白さは危険とさえいえた。対戦ポピュラスは人間の潜在的な残酷さを引きずりだす。対戦専用の身の毛もよだつような技も多数開発され，Oh!X編集室でもスタッフが連日泥沼のような対戦を繰り広げていた。祝一平対西川善司の死闘はいまも僕の記憶に生々しく残っている。

そして，ポピュラスⅡ

物語は例によって全能の神ゼウスの好色から始まった。神であるゼウスを父に，人間を母にもつ主人公は，優れた力をもちながら神々の仲間入りを果たせない。

そこで主人公は，自分の力を神々に認めさせるべく，ゼウスを頂点とするオリンポスの神々に戦いを挑んだ。目標はゼウスを打倒すること。息子は父を超えることができるか。

……というのが「ポピュラスII」のストーリーということになっている。しかしいったんゲームを始めたら，このストーリーは忘れてよろしい。要は目の前の敵を完膚なきまでに叩き潰せばいいのである。

「ポピュラスII」は基本的に前作の「ポピュラス」と同じゲームである。しかし，ある部分は洗練され，ある部分は強化され，前作の欠点はほぼ取り除かれた。それをこれから見ていくことにしよう。

充実した神業

「ポピュラスII」は2モノにふさわしく，ゲーム内容がかなり強力になっている。まず，使える神業の種類が大幅に増えた。ここでまとめて紹介しておこう。前作「ポピュラス」にも同様の神業がある場合は併記する。

「ポピュラスII」の神業は数が多いので6つのカテゴリーに分けられている。神業を起こす際は，まずカテゴリーを選んでそのなかから使いたい神業を選ぶことになる。ここはワンアクションで神業を選択できた前作に比べて，若干操作性を落としたところであり，慣れるまではとまどうことになる。どの神業がどのカテゴリーに属するかは，わかりやすい分類になってはいるものの，ある程度は覚えないといけない。

これらの神業は前作と同じく，面の設定によって使えたり使えなかったりする。征服モードでは，この設定が面の難易度を決定する重要な要素となる。

以下に紹介する「ポピュラスII」の神業のうち，僕にとって特に印象的なのは地震，嵐，津波，火の雨，火山，泉，そしてトロイのヘレンである。

人の神業カテゴリー

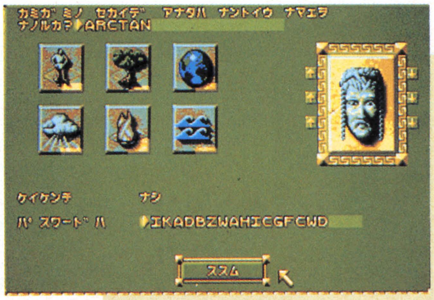
○土地の造成（前作と同じ）

とにもかくにも土地造成である。素早く領地を拡大して，マナを効率よく蓄積するのが勝利への道である。

○マグネット配置（前作ではシンボル配置）

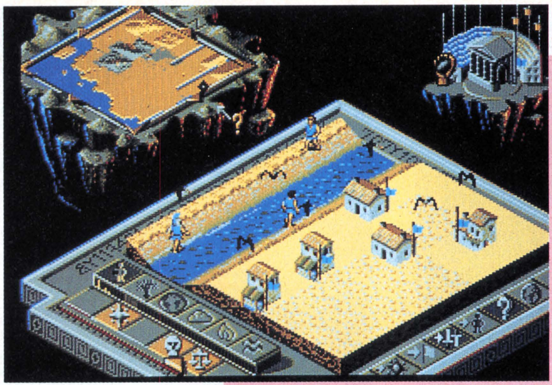
民衆を導くためのシンボルは，見た目がちょっとだけ妙なものになっているほかは，その役目も動作も，前作とまったく同じである。

○ペルセウス（前作では騎士）



顔を変えたり，経験値を振り分けたり

リーダーをペルセウスに昇格させる。騎士は「ポピュラスII」では「ヒーロー」と呼ばれ，種類も増えている。ヒーローは騎士と同様，リーダーを昇格させて作る殺戮マシンである。ペルセウスは「ポピュラスII」のなかでは標準的なヒーローである。


疫病流行って，カラスがカァー

ハルマゲドンで一気にケリをつける

○疫病

敵を疫病に感染させる。疫病は相手のマナの蓄積を阻害する働きをもつ。いやがらせに使えそうな神業。疫病を敵の建物または民衆に感染させると，民衆が繁殖するにつれて敵陣全体に蔓延する。蔓延を防ぐには，疫病に感染した人を殺し，建物を破壊するほかない。疫病に感染した人や建物の上空には，カラスが不気味な鳴き声をあげながら飛んでいるのでわかる。疫病はその性質上できるだけ早く使うのが効果的であるが，マナの消費も大きいのでおいそれとは使えない。

○ハルマゲドン（前作と同じ）

ハルマゲドン（最終戦争）を宣言する。「ポピュラスII」でも勝利条件は敵が全滅することである。民を誘導するなりヒーローを送り込むなりして，敵を片端からやっつけていってもいいのだが，このハルマゲドンを起こして総力戦にもちこむと早く決着がつく。「ポピュラスII」ではハルマゲドンを起こしたあとでも，なぜか神業が使えるようである。ハルマゲドン後に敵の足元に沼をバラまくような，卑怯な真似も可能なようだ。

植物の神業カテゴリー

○森

木を生やす。木はそれだけでは無害である。しかし森には延焼というおそるべき使い道がある。敵陣に木を密生させておき，その端に放火すると，またたく間に火は森全体を焼き尽くす。このとき，森の中にある建物に引火するので，効率のよい破壊活動が可能になる。

日本語はわかりやすいけど，くすぐったい

○土地の再生

火山などで荒れた土地を再生する。「ポピュラスII」の火山は強烈で，一度火山が噴火した土地は一面火山岩で覆われてしまい，土地をいくら造成しても元に戻らない。この神業で再生して初めて建物を建てることができるようになる。

○沼（前作と同じ）

沼をバラまく。どんなに強いキャラクタでも，沼に落とせば一瞬で殺せる。前作「ポピュラス」で，もっとも直接的かつ強力な攻撃方法であった。2では，地震にそのお株を奪われはしたが，依然として強力な攻撃手段。

○カビ

カビをバラまく。その性質はひとことでいえば増殖する沼。敵陣の近くに並べておくと，そのうち繁殖していく。その上を通った人間は飲み込まれてしまう。カビはライフゲームのアルゴリズムで増殖するとかしないとか。

○アドニス

リーダーをアドニスに昇格させる。アドニスはヒーローの一種である。敵と戦うたびに分裂し，かなりの勢いで破壊活動を展開する。分裂したアドニスたちがわさわさと襲ってくる姿は恐怖である。

土の神業カテゴリー

○道路

道路を作る。民衆はこの道路に沿って歩きやすいという傾向をもっている。沼などの罠を仕掛けておいて，そこに誘導するのにも使える。

○城壁

城壁を作る。城壁は敵の侵入を防ぐ。敵のリーダーを殺してシンボルを城壁で囲い込んでおけば，敵はシンボルまで到達できず，リーダーを（つまりヒーローを）作れなくなるという応用技がある。

○地震

地震を起こし地割れを作る。この地割れは強力。どんな強いキャラクタをも飲み込むところは沼と同じだが，ランダムにバラまかれる沼と違って，地割れは線になっているので出現位置をコントロールしやすく，沼よりも確実に敵を捉えることができる。

○土地の隆起（前作では火山）

土地を隆起させ，同時に岩を発生させる。前作の火山に相当する。なお，「ポピュラスII」の火山は，噴火して溶岩も流れる本当の火山である。

○ヘラクレス

リーダーをヘラクレスに昇格させる。ヘラクレスはヒーローの一種で，怪力を誇っている。

気の神業カテゴリー

○雷

雷雲を発生させ，雷を落とす。雷の落ちた建物は燃え，人間は感電する。それほど威力はないが，弱いキャラクタなら簡単に倒せる。

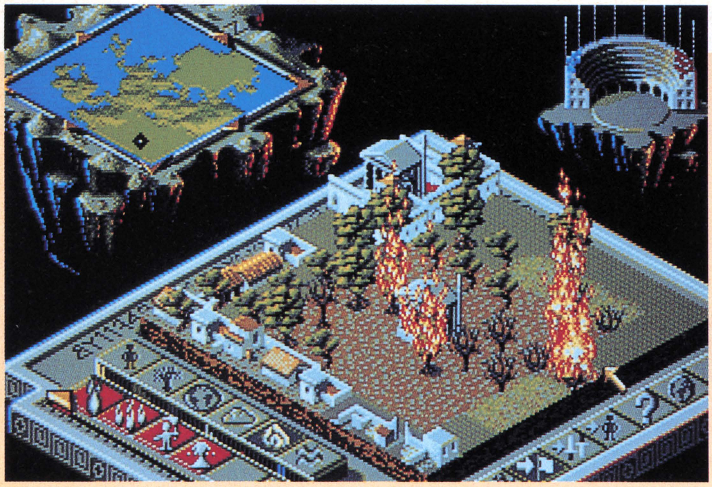
○竜巻

竜巻を発生させる。竜巻に巻き込まれた人間は遠くに吹き飛ばされ，目を回すためか知能レベルが落ちるとされている。

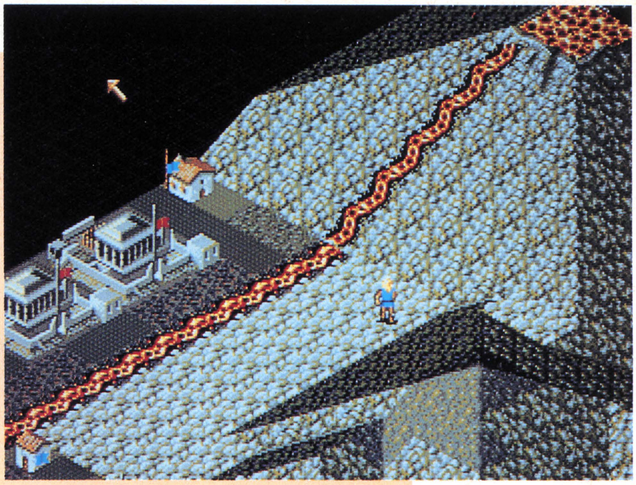
○嵐

嵐を起こす。雷雲の大規模なもので，長時間にわたって敵陣に雷を落とし，焼け野

雷が落ちたところは火がつく



木を植えたあとで放火するとは気の長い



今回のは正真正銘の火山。マグマも流れ出す

原を作る。

○強風

強い風を起こす。この風は建物や人間やシンボルまでも吹き飛ばす。

○オデッセウス

リーダーをオデッセウスに昇格させる。オデッセウスはヒーローの一種で，足が速い。通常のヒーローの倍の速度で走り回り，破壊活動を繰り広げる。

火の神業カテゴリー

○火柱

火柱を立てる。この火柱はゆっくりとランダムウォークで動き回るが，なかなか建物にヒットしないので，森と組み合わせて使うのが効果的。延焼の火付け役として使うのである。

○火の雨

火の雨を降らせる。この焼夷弾は火柱より広い範囲を焼き払う。

○火山

火山を噴火させる。火山の項上からは溶岩が流れ出して，人や建物を焼き払う。溶岩は小さな丘くらいなら乗り越えて流れていく。火山が噴火したあとの土地は火山岩の荒れ地となり，土地を再生しないかぎり建物を建てることができない。

○アキレス

リーダーをアキレスに昇格させる。アキレスはヒーローの一種。火をつけて回る。

水の神業カテゴリー

○火山岩

海の上に火山岩の道を作る。土地を造成しても火山岩は消えないので，敵の土地の近くに置いて，いやがらせに使う。

○渦潮

海に渦潮を放つ。この渦潮は海上をランダムウォークしながら陸地を削り取る。敵の陸地の周囲にいくつも渦潮を放つと事実上造成ができなくなるので，これもいやがらせに使える。

○泉

泉をバラまく。この泉の聖水には，敵味方を転換する力がある。この泉に落ちた敵は味方になってしまう。強い敵が襲ってきたときなど，こちら側に寝返らせられる。当然，泉に味方が落ちると敵になるので注意を要する。

○トロイのヘレン

リーダーをトロイのヘレンに昇格させる。トロイのヘレンはその魅力で男どもをたぶらかして地獄に叩き落としたといわれる伝説の美女。「ポピュラスII」でもヘレンはもっともおそるべきキャラクタである。敵陣をしゃなりしゃなりと歩き回り，民衆をハープの音色でたぶらかし，民衆は本来の任務を忘れてふらふらとヘレンのあとについていく。そして，ヘレンは頃合を見計らって，連れ去った民衆を抹殺してしまうのだ。自分が使うぶんにはいいが，敵に使われると厄介なキャラクタである。早々に沼や地震で殺すか，泉を使って味方にして利用する（僕はこの技を「ヘレン返し」などと呼んで重宝している）のが得策。

美しい声で誘惑し，まとめて殺すヘレン

○津波

海上に津波を発生させる。津波は陸地を削り取り，人や家を飲み込む。海の向こうから陸地に襲いかかる津波の眺めは壮観である。中途半端な高台も削り取るが，堤防を作っておけば防ぐことができる。

新システム

前作の「地震」と「洪水」は姿を消した。ポピュラスの達人だった人ならわかると思うが，この2つの神業はほとんど役に立たなかった。

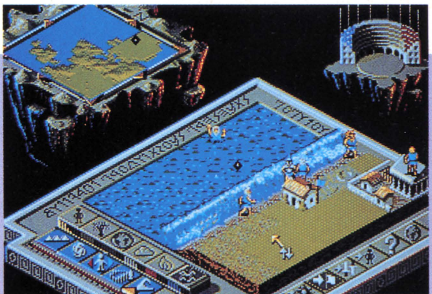
地震は建物を壊すが人を殺せないので，かえって敵の民衆を拡散させ，結果として敵の繁栄を助ける結果を招いていた。「ポピュラスII」の地震は同名だがまったく別の効果をもっている。あまりに強力すぎて，かえってバランスを崩している面があるくらいだ。

洪水は一瞬にして全世界の水位を上げるもので，マナを大量に消費するわりには実入りの少ないものであった。実戦で使われているのもほとんど見たことがない。

さて，「ポピュラスII」で初めて導入されたシステムもある。

○経験値

RPGの必須アイテムとされる経験値が，リアルタイムシミュレーションである「ポ



津波がくると海岸が削られ，人が溺れる

ピュラスII」に導入されている。

征服モードで勝つと，その勝ち方が評価され，得点が与えられる。それが経験値につながる。評価方法は不明だが，初期設定(使える神業の種類や人口)の有利/不利が評価に大きく影響しているようだ。要するに，不利な状況で勝てば，それだけ高い経験値を得ることができる。

経験を積むと神業の効果がより強力になっていく。といってもその変化は微々たるものである。

○スプログ

ポピュラスの基本は人口を効率よく増やすことにある。人間は建物の中で繁殖し，それが大きな建物であるほど速く繁殖する。そして人数が建物の容量を越えたときに，あふれた人間が外に出てきて空き地に新しい建物を建てる。

ただ，こうした自然増加を待っていたのでは，領地を素早く広げることは不可能だ。このため「ポピュラス」では，建物のそばに小さな丘を造って，建物を一時的に小さくして人間をあふれさせるという技が必須のものとなっていた。しかし，コンピュータはこの技を使わず，民衆の自然増加に任せていた。

そこで「ポピュラスII」では「スプログ」という操作が出てきた。建物を直接クリックすれば，人間を強制的に外に出せるのである。これだと地形も変わらない。

スプログの恐ろしいところは，コンピュータも積極的に利用するところである。ほとんど並列処理なみの勢いで建物を片っ端からスプログし，出てきた民衆が建てた家の周囲を整地し，またたく間に城にしてしまう。知らないうちに，敵の人口が2倍3倍になってしまうこともしばしば。

スプログにかぎらず，コンピュータのマメさは恐ろしい。沼や地震や火山を執拗に放ってくるのでこちらはてんやわんやだ。ゆめゆめ速いマシンで遊ぼうなどと考えてはいけない。

○造成コマンドの制限

しばらく前，「ポピュラスII」のオリジナル(AMIGA版)発売直後にも書いたが，「ポピュラスII」では，「敵の陣地に対して造成コマンドが使えない」ように設定できる。作者のピーター・モリニューが，日本人プレイヤーに対して挑戦しているような気がしてならない。彼はかつて来日したおりに対戦ポピュラスをやったのだが，日本側が当然のように「手作りの山」を多用したために，あっさり負けてしまった。

手作りの山とは敵陣内で造成コマンドを使って山を作ること。営々と築き上げた建物は一瞬にして崩れ落ちてしまう。少ないマナの消費で敵陣をズタズタにできるので，対戦ポピュラスには必須の戦法だったが，作者自身はまったくその戦法を想定していなかったようだ。

さんざんやった僕がいうのもなんだが，たしかに手作りの山戦法はやっていて空しくなる。手作りの山は強力だが不毛である。戦略もなにもない。なにより美学がない。ただ勝つだけでなく美しく勝つことも大事なような気が最近している。

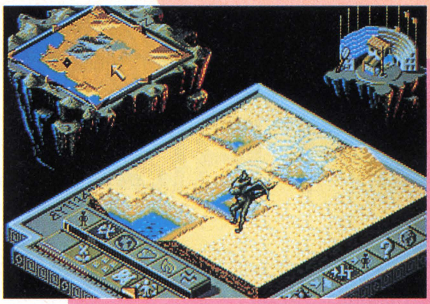
面白くなったかな？

「ポピュラスII」をひと目見てまず気づくのは，画面の質感の違いであろう。前作では，とことこ歩くかわいいキャラクタが殺戮を繰り広げるという妙な違和感が印象的であった。その後，「パワーモンガー」を経て，「ポピュラスII」のキャラクタはもう少しリアルなものになったようである。神業で出現するものや人間たちなど，あらゆるものがアニメーションする。とにかく芸が細かい。

「ポピュラス」は“II”になり，全体にゲームが派手になった。神業はその視覚効果にふさわしく，多様かつ強力になった。

神業が強力というだけでなく，マナの蓄積も速いようだ。したがって攻撃も派手にできるが，攻撃を受けてもダメージの回復が速い。なすすべもなくボコボコにされるという悲惨な負け方は少なくなり，しぶとい抵抗が可能になった。このため戦闘がより長期化する傾向を見せている。上級者どうしの対戦などは，きっと泥沼になるに違いない。

火系統の神業は，今回新登場のものばかりだが強烈である。火の雨を降らせたり火山を噴火させたり。森を併用して敵陣を火の海にするという複合技もある。敵の建物を紙くずのように燃やしていくのはアブナイ快感である。ポピュラスの快感は沼にハメることにあったが，「ポピュラスII」の快感はどうやら放火にあるようだ。



モンスターもいろいろと出る

経験値システムには賛否両論あるかもしれない。僕はといえば，保守的かもしれないが「否」側である。

理由のひとつは，対戦でイコールコンディションの戦いがしづらくなるということ。もうひとつは，「ポピュラス」では経験を積んで強くなるのはプレイヤー自身であるということ。的確な状況判断と適切な戦略，そして素早いマウスさばきが「ポピュラス」に勝利するために必要な力である。そしてそれは場数を踏むことによってのみ身につくものである。むしろ，経験値稼ぎを強要して得た数値パラメータでしかプレイヤーの技量を表現できないRPGのシステムのほうが浅薄なのではないか？　そんなものを真似る必要などないね，といいたい。

早く対戦したい

「ポピュラスII」は「ポピュラス」ほどのインパクトはないが(2モノの宿命)，ポピュラスよりスケールアップしており，なおかつ面白い。

まだ編集スタッフのなかには，僕以外に「ポピュラスII」をやり込んだ者がいないので対戦をちゃんとやっていない。

最終的な評価は対戦のあと，ということになるだろう。

ポピュラスII

海外作品ならイマジニアっすね

オリジナルはいうまでもなくAMIGA版。国産マシンへの移植はX68000版が最初。アセンブラで書いてあるという話だから，これはむしろ当然(前作はC言語)。

移植のデキであるが，例によって「イマジニアの仕事」ということで想像がつくことと思う。グラフィックは完璧にオリジナルどおり。速度はオリジナルより心なしか速いくらい。逆にX68000 XVIだとコンピュータの反応が速すぎて征服モードで歯が立たない。10MHzで遊ぶのが快適。操作性もよい。

効果音の処理はあいかわらず苦しそう。しかし，ゲームを進めるのに支障のないくらいには効果音を再現できている。

メッセージ類はすべて日本語に訳された。こういっちゃなんだが，いまどき全部カタカナというのもちょっと……。BGMは日本版オリジナルであることを強調しておく。

総合評価

	0　　5　　10
グラフィック	★★★★★★★★★
その移植度	★★★★★★★★★★
効果音	★★★★★★★★
その移植度	★★★★★★★
BGM	★★★★★★

VTRはヤンエグの香り

Urakawa Hiroyuki
浦川 博之

誰でも一度は，“会社の経営者になってみたい”と思ったことがあるのではなかろうか。しかし，実現するのはむずかしい。多大なお金や人が絡んでくるから。とりあえずは，このゲームで適性を見てからにしたほうがいいだろう。

就職活動。理系の学生ってのは教授の推薦なんかで簡単に決まっちゃうことが多いらしいけど，文系の学生にはなかなか大変な作業だ。スーツを着込んで，あちこちを駆けずり回り，汗だくの顔をニッコリゆがめながら，「御社が第一志望です」。このセリフが10回目だったりするから油断ならない。どうせ相手も信じてないけどね。

活動の第一歩は企業にアポイントを取ることから始まる。

プルルルルル。

「はい，ポニー人事部です」

「あの，羽田大学のうりゃかわと申しますが，御社の採用のご予定は……」

「内定です」

「は？」

「だから，内定です」

「電話だけで？」

「あなたには社長に就任していただきます」

「知ってのとおり，初の家庭用VTR発売から1年，VTR市場は規格が乱立し，本格的な普及を阻んでいる。君の力でこの規格戦争を終結させ，4000億ともいわれるこの市場を手にしてもらいたい。健闘を祈る」

いきなり口調が変わるな。ちょっと待った，どこにそんな無茶苦茶な話が。

「君の仕事のサポートとして有能な美人秘書をつけよう」

「任せてください」

X68000用 3.5/5″2HD版2枚組 12,800円(税別)
光栄　☎045(561)6861

ヤンエグ新人研修を受ける

社長室の椅子に居心地悪く座っていると，ドアをノックする音がした。

「社長秘書の令子です」

金髪の女性。いかにもポニーって感じ。

「新入社員で社長のうりゃかわです」

「それではさっそく社長としての勉強を」

「一緒に銀座に飲みにいく」

ぱこーん！　有能秘書のスリッパ攻撃。

「まずVTR業界の現状について勉強していただきます。これをご覧ください。これはVTRのマーケットを地図にしたものです。右奥にいくほど低価格，左奥にいくほど高性能であることを表しています」

「我がポニー率いるBE規格のビデオの勢力は緑で表されています」

「わりと高性能路線なんじゃないの」

「そのとおりですわ。さすが社長」

「ふふふーん」

ふんぞりかえる社長だ。

「当然ですが，地図の奥に位置する製品ほど多くのニーズをカバーできるわけです。このVTR業界でシェア80％を握り，価格と性能ともに最高の製品を発売していただくのが社長の使命です。2年間赤字ならすぐさまクビですのでお忘れなく」

「いきなり社長になって，2年でクビ。普通の人の20倍ぐらいのペースだ」

「では経営会議のほうへ」

会社は主に3つのセクションから成り立っている。営業部，製造部，研究所だ。研究所で開発した製品を製造部で作り，営業部の人たちが売る。その進行状況を聞くのが経営会議で，各ターンの前に行われる。

経理からの収益報告，営業からの売り上げ報告，製造からの工場稼働率報告，研究所からの開発状況報告。

「おわかりになりましたか，社長」

「とりあえず儲かってるみたいだね」

不安だわ。社長秘書令子は思わず眉根を寄せた。

ヤンエグ経営に乗り出す

いよいよ社長業の開始だ。社長業の第一歩は，社員の声を聞くことから始まる。命令を出さずに待っていると，各部署のおしゃべりが画面にチラチラ映るのだ。

製造「AIDA 2 BAS（製品名）は，作ってるわりには売れないなぁ」

営業「広告のノウハウが足りないなあ」

開発「この研究所の人員じゃ，新製品も望み薄だね」

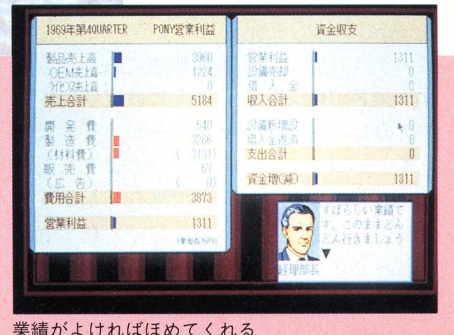
「令子さん，なんかえらく社員の評判が悪いんですけど」

「社員の声に応えるのが，いい社長になるための条件ですわよ」

うりゃかわ君は社員の声に応えることにした。技術のポニーだからして，研究所の改善から取り組む。研究員を増員して，予算を増やす。技術力が上がった。

「技術力がつくとどうなるの？」

VTRのマーケット地図



業績がよければほめてくれる

「短い期間で新鮮度の高い製品が作れるようになります。新鮮度が高ければ売り上げも伸びますし，モデルチェンジまでの期間を長くすることができますわ」

我ながら賢明な処置である，と胸を張る。製造部門にも予算を多く回して合理化を進めさせよう。各部門への指示が終わると，広告戦略を練るステージが待っている。

「ちわー，白鳳堂です」

「おう，白鳳堂さん，広告を頼むよ」

「BEブランドの宣伝と，ビデオの製品宣伝と２種類ございますが」

「知名度を上げたいから，製品宣伝」

「ご予算に応じて販促物配布からTV番組提供までございますが」

「じゃあ，この全国フェアってやつ」

「9700万円いただきます」

「高いぞ，おい」

「広告のおかげで伸びる売り上げに比べれば安いもんですよ」

広告開始。

“製品広告は失敗しました”

「てめー，白鳳堂！」

「あでで，おたくの営業部員が足りないとこういうこともあるんです，ぐるじい」

社内に指示を出し，広告を打つと，いよいよ今期の製品売り上げが発表になる。

ピピピピピピピ。

「お，売り上げが伸びた」

「よかったですわね，社長」

「新製品発表まで，もたせられるかな」

「それには営業の強化が必要かと」

次のターンでは営業の強化を図ろう。結構お気楽に考えているうりゃかわ君だ。

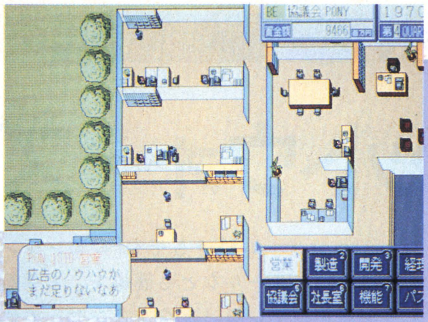
ヤンエグ長期戦略を練る

そんなこんなで数年が経つ。各社ともに新製品を発売し，高性能，低価格，新鮮度の高い機種から売り上げを獲得していく。

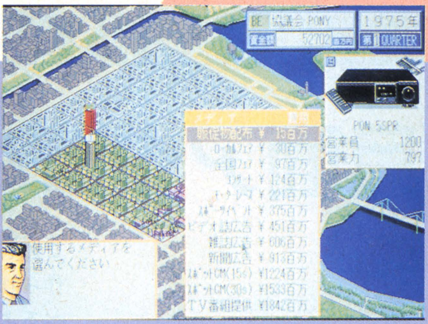
うりゃかわ社長率いるBE規格は，徹底した高性能路線をとって成功していた。性能，価格ともにそこそこの市場は乱売気味で台数が伸びない。ライバルより安い機種を作れば売れるかもしれないが，製造原価が高くて儲けにならないのだ。

しかし，この世界はコストパフォーマンスだけがすべてではない。時代の流れというものがある。レンタルビデオの普及，高画質テープの登場，４年に１度のオリンピック。それぞれが市場に影響を与える。

ビデオに求められる機能も当然変わってくる。リモコンがついてないと売れないとか，バーコード予約のある機種に人気が集まるとか。時代のトレンドを読み，先手を打った経営をしなければならないのだ。



社員の声は尊重しなければ



各種メディアも利用できる

加えて，市場が大きくなってくると自分の会社だけでは生産が追いつかなくなってくる。このゲームでは，なぜか自社規格の製品は自社でしか開発できないので，ライセンス生産やOEM供給をしてくれるパートナーを探さないとシェアが伸びないのだ。

うりゃかわ社長は本人の性格が災いしてか，同業他社に嫌われまくっていた。

「なぜだろう。SHARKからMIBISHIから，他社の信用度がみんなゼロだ」

「営業の姿勢が強引すぎるという批判が出ていますわよ。それにパートナーづくりをしてこなかったために，ポニーは利益を独占しているという噂もたっています」

「ま，いいや。我が社が市場を制すれば向こうから頭を下げてやってくるさ」

すっかり態度がでかくなっている。それでいいのか，うりゃかわ。ゲームの終了条件にはVTR市場を4000億円の規模にするというのもあるのだぞ。ポニーだけで4000億も売り上げられると思っているのか。

ヤンエグ勝利の美酒に酔う

社長に就任して12年。ついに使命をまっとうすることができた。4000億円の市場を開拓し，価格40,000円，性能プロ級の製品を作り（赤字にしないようにするのは大変だった），市場のシェア80％を握る。

結局，他社に頭を下げて回り，自社規格の製品を作ってもらったのだった。市場を独占しすぎたために，「ウチでは生産が需要に追いつきそうもないですなぁ」といわれ続け，そこを口説いてファミリーに入ってもらうのに多大な時間を要した。

ポニービルのラウンジで夜景を眺めながら，ひとりくつろぐ，うりゃかわ社長。研究所の所員を引き抜かれたり，工場のラインがストップしたりしたのも，いまとなってはいい思い出さ。ライバルの規格は時代の読みを誤ってみんな撤退した。

「社長」

令子が入ってきた。

「令子くん。君には世話になったな」

さりげなく肩に手を回したりして。

「社長。実は新入社員が入社することになりまして」

「ほう。それで？」

「新入社員が社長に就任する慣例に従って，社長には退任していただくことに」

「へ！」

就職活動の第一歩はアポイントメントを取ることから始まる。

プルルルルル。

「はい，ベクター人事部です」

「御社の採用のご予定は……」

光栄流に練り上げられた企業戦略の世界

会社経営というむずかしいテーマのわりには，簡単に理解できて遊びやすい。命令を出しては，それが売り上げにどう結びつくか確かめるという作業の繰り返しで，ついついハマってしまうという感じ。新製品の売れ行きを見る瞬間などは，ついドキドキして見入ってしまう。

このようなテーマをゲームとしてまとめあげる力には，さすが光栄だというしかない。しかも，ほかの歴史シミュレーションに負けない面白さがあると断言できる。各メーカーの人間の絵は「いかにも！」って感じで笑えた。

しかし，光栄ゲーム全般が抱えている問題はやはりこのゲームにも存在する。それは情報の参照が面倒臭いことと，操作性が悪いこと。はっきりいって，ツリー型のコマンド構造はもう古い。製品なら製品の全体的な情報を一発で全部見られるようにしてほしいものだ。

あいかわらずX68000らしさが感じられないこともある。ジャズを採譜したBGMや，アメコミっぽい人物のグラフィックにはセンスを認めてあげるから，全体にそのセンスが行き渡るように頑張ってほしいもんである。

総合評価	0　　5　　10
グラフィック	★★★★★★★★
BGM	★★★★★★★
操作性	★★★★★★★
熱中度	★★★★★★★★★
コンピュータの強さ	★★★★★★
スピード	★★★★

月面アクロバットウォーズ

Kageyama Hiroaki
影山 裕昭

このシミュレーションゲームはそもそもPCエンジン用に作られたもの。ゆえに，ルールなども簡単で，お手軽に遊べるような工夫がなされている。アクションゲーム感覚でシミュレーションを遊んでいるという雰囲気なのだ。

中東で湾岸戦争が勃発した直後，戦争がテレビゲーム化しているとの報道があった。ミサイル攻撃はジョイスティックで照準を操作し，攻撃目標物を正確に破壊することができるピンポイント方式である。テレビで見る映像は，まさにゲームの画面のようであった。湾岸戦争の報道では無残な死体をテレビ画面に流すことが少なかった。連合軍はクリーンな戦争だと強調していたが，きれいな戦争なんてあるわけがない。

別に私は平和主義者ではない。湾岸戦争の話を持ち出したのは，同時期の日本でウォーシミュレーションゲームが好調な売れ行きだったことを思い出したからである。

マニュアル不要

この「ネクタリス」はPCエンジンからの移植である。グラフィックは（おそらく）ベタ移植で，解像度は256×256ドット。アクションゲームならともかく，シミュレーションゲームでこの解像度ではガッカリだ。X68000のグラフィック機能を存分に生かしてほしいものである。

気を取り直してゲームの解説をしよう。ゲームは近未来21世紀が舞台となっている。人類は月面に進出し，採掘した資源を地球に供給していた。地球ではその資源を巡って大国が権利を主張し，そのなかのひとつ，ガイチ帝国がついに軍事力を発動させ，月に軍隊を送り，ほぼ全土を占領してしまった。ガイチ帝国は月面で地球攻撃の最終兵器MOAの発進準備を進める。これを知った月の反乱軍は，監禁されていた収容所を集団脱走し，MOAの発射基地であるネクタリスを目指した。プレイヤーは反乱軍の総指揮を取り，行く手を阻むガイチ帝国を粉砕してネクタリスへ侵攻し，MOAの発進を阻止することが目的である。

ゲームではいくつかの戦場が用意されており，ひとつの戦闘に勝利すると次の戦場に進めるようになっている。戦闘途中のゲームセーブはできないが，「ネクタリス」では戦場名が継続プレイに必要なパスワードとなっている。スタート時に最後にプレイした戦場の地名を入力すると，その戦場から再開することができる。

また，タイトルメニューにはMANUALという項目がある。これは何かというと，マニュアル＝説明書である。ゲームの流れや目的，コマンドの説明などがビジュアルと文章で行われ，猿でもわかるのではないかというくらい丁寧な説明がされる。

シミュレーションというと分厚いマニュアルがつきものという印象が強いが，「ネクタリス」に関してはそのことを忘れてもらってかまわない。難点をいえば，このマニュアルはゲーム中には参照することができないことだろうか。

X68000用　5"2HD版　7,800円（税別）
システムソフト　☎092(752)5262

ネクタリスを目指せ

「ネクタリス」の戦場マップ画面にはあらかじめ敵味方の部隊（ユニットと呼ぶ）が配置されている。自軍のユニットは青色，敵のユニットは緑色で表示される。各マップには敵味方とも必ずひとつの収容所を持っている。勝利条件は敵の収容所を占領するか，敵部隊を全滅させること。この逆が成立すれば自軍の敗北である。

戦場に自軍の工場があれば，戦闘で戦力が低下したユニットを工場で補給することができる。マップによっては中立工場がある。中立工場を占領すると自軍の工場として使えるようになる。操作はPCエンジンからの移植ということもあってジョイスティックでしか遊べないんじゃないか……，なんて心配をしたもんだが，マウスでの操作もできるようになっていた。

キーボードがなくても遊べるかというと，敵に降伏をするときだけはキーボードからの操作が必要である。もし誤操作を防ぐためにこうしたのであるなら，なかなか考えられているといえよう。

ゲームはターン方式で進行する。ターン方式というのは，プレイヤーとコンピュータが交互にユニットを移動したり，敵に攻撃したりしてゲームを進めていく方法である。通常，マップ画面にヘックスは表示されない。ユニットを選択すると移動可能な範囲でヘックスが表示される仕組みである。最近はこういうシステムが主流らしいが，初めて見たのでちょっと驚いた。

攻撃力や防御力はユニットが配置されている地形によって変化を受ける。敵のユニットが道路にいるのと，山の中にいるのと

戦場全体を確認して動きを予想する

戦闘シーンはもちろんアニメーションする

では防御力が違ってくる。当然山の中にいたほうが防御力は高い。支援効果，包囲効果も戦闘結果に影響してくる。これは敵ユニットの周りを自軍のユニットで囲むと，敵の防御力を減らし，自軍の攻撃力を高めることができるというもので，防御力の強い敵を攻めるとき，防御力の弱い自軍を守るときに有効な方法となる。

すべてのユニット性能はゲーム中でも確認できる

工場に格納されたユニット群。頼りは航空機

GALOISでの攻防

ガイチ帝国と9回目の衝突が起きたGALOISは，スタート時点で敵ユニット数が自軍をうわ回った初めての戦場である。中央の険しい山脈を挟んで左に自軍の収容所，右に敵の収容所が配置されている。

保有する工場はそれぞれひとつで，中立の工場が4つあることが確認された。自軍の近くの中立工場は占領が比較的簡単にできそうだ。敵軍の近くにもひとつの中立工場がある。これは間違いなく敵の手に落ちるだろう。中央の山の麓にある2つの中立工場の占領がカギとなりそうだ。

我が軍には間接攻撃を仕掛けるユニットがひとつしかないが，敵は4つ持っている。これを踏まえて我が軍は，最初は航空機で敵の間接攻撃ユニットを集中攻撃して壊滅的なダメージを与え，中盤で中央の中立工場を占領し，最後には空と地上から敵を一斉攻撃する。初期の段階では敵の対空砲火も相当厳しいものが予想される。地上から敵対空ユニットへの攻撃を行い，航空機を援助することに決定した。

我が軍は中央の中立工場に向けて進撃を開始した。前線に重戦車のグリズリー，続いて対空ミサイルランチャー装備のラビット，中型量産戦車のバイソンを配置。後方には防御力が高い重戦車アルマジロ，間接攻撃のナスホルン，歩兵ムンクスを爆撃機から保護するため対空車両シーカーとホークアイを援護につけた。

工場からはトラックに歩兵を乗せて，別の中立工場の占領を目指す。対地爆撃機ジャビイ，対空攻撃機ファルコを発進，アルマジロも最前線の防衛のため出撃させたところで自軍のフェイズを終了した。

続いてコンピュータのフェイズだ。最前線にガイチ軍の最強重戦車モンスター，後続にシーカー，ラビットと戦車を連ねて配置してきた。対空車両の間近に航空機を配置してくるあたり，なかなか考えているようである。後方には間接攻撃が可能なホークアイ，ロケットランチャー装備のエストールで前線に攻撃を仕掛ける我が軍に重砲を向けている。1ターンが終了した。

2つある中立工場のうちのひとつは敵の戦車で包囲されてしまった。現状で戦力の劣る我が軍としては，戦闘となっては不利である。ここは空いている中立工場の占領に全力を注ぎ，1ターンでも早く最前線に工場を確保することにしよう。最前線で戦力の補給が可能になれば，戦闘で低下した戦力を素早く回復することができる。

とはいえ，工場の占領は歩兵やバイク兵でないとできない。歩兵は航空機から攻撃されると全滅することにもなりかねない。歩兵を失えば工場の占領ができなくなるので，この歩兵はなんとしても生かしておかなければならない。「ネクタリス」では地上ユニットと航空ユニットであっても同一のヘックスに属することは許されていないから，歩兵を囲む6つのヘックスに自軍のユニットを配置してしまえば，敵の直接攻撃を受ける最悪の事態だけは防げる。

初めての敗北

3ターン目に敵は2つの中立工場を占領した。我が軍は4ターンめにやはり2つの中立工場を占領した。保有する工場の数は同じだが大きな落とし穴があった。中立工場は1～6個のユニットを抱え込んでいて，占領と同時にそれらのユニットを自軍のものとして使うことができる。敵が占領した工場には重戦車や間接攻撃ユニットが豊富にあったが，我が軍では2機の航空機を除いて，ほかは防御力の低い貧弱なユニットが多かったのである。地上ユニットは防御力が低いうえ，数で敵に圧倒されている。

なかばヤケクソ気味に航空機で敵地上ユニットを攻撃。わずかに敵の戦力を減らすことに成功したものの，7ターン目に大事な航空機をすべて失ってしまった。敵の攻撃に押されているのが目に見えてわかる状態になっている。頑張って敵のユニットを攻撃しても，1機でも生き残りがいれば，工場に戻って8機に回復してしまうのだから，らちがあかない。10ターン目には最前線の戦いを回避したガイチ軍の大型空中輸送機が我が軍の収容所に歩兵ムンクスを輸送し，収容所は敵の手に落ちた。

MOAは予定どおり地球に向けて発進，地球はガイチ軍に降伏した。

グラフィックが悲しいかな

本文中では触れなかった，コンピュータの思考能力について話そう。前半はマップが小さいし登場するユニット数も少ない。コンピュータのユニットの動かし方を見ていると，これがなんともマヌケであった。これが後半のマップになると，前半のようなバカな行動が目につかなくなる。はたして最初はワザとマヌケに動かしているのか。それはわからない。ゲーム全体を通すとそれほどバカでもないので，いい加減な戦術では負かされてしまう。

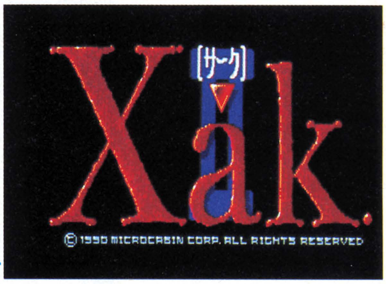
敵の思考時間も短く，待たされてイライラすることもない。初心者向けであるためとっつきやすいが，そのぶん大戦略シリーズなんかと比べると戦術の数は少ない。

あと，1ターンの終了を選択したときに，いきなり敵のフェイズに移るのは気に入らない。終了コマンドがよく使うコマンドの近くにあるので，誤って選択する可能性は高い。終了時には確認を求めるのが常道だろう。

ユニットに弾切れはない。燃料切れもない。工場で受けられる補給は，1機でも残っていれば1ターンで1ユニットのMAXである8機になる。これは自軍のユニットを工場に入れたときにはありがたい仕様だが，敵のユニットにやられると涙チョーチョーである。

総合評価	0 ― 5 ― 10
グラフィック	★★★★
操作性	★★★★★★★★
戦略性	★★★★★★
X68000らしさ	★★★★

For the human over the one.



X68000版「サーク」はとっくにプレイしたけど，「サークⅡ」はまだ出ないのかなあ。他機種ではずいぶん昔に出たのに。そう思っていた人には朗報。今回もいろいろな頼みごとを受けて，人のいいヒーロー役を演じてくださいな。

Taki Yasushi
瀧　康史

ついに「サークII」が，X68000でも遊べるようになりました。前作「サーク」はX68000向けにアレンジされていて，512×512モードの画面でプレイできたので，「サークII」も絵がきれいになっているかなとちょっと期待。そう思いながら，ディスクをセットして起動。あらら。まるっきりの640ドット×200ラインのベタ移植でした。でも，スプライトも使わないベタ移植のわりにはスピードが速いなあ。

起動するといきなり選択肢が出て，主人公の名前を変えるか，オープニングを見るか，ゲームを始めるか，を選べます。

名前を変えてもいいのですが，せっかくラトク・カートという名前があるんだから，私はこれを利用しました。

戦闘システム

「サークII」は前作同様，いろんな面で立体を意識したアクションロールプレイングです。VRシステム2だとかというのを使っていて，これによって，当たり判定なども3D化しているようです。そのために，空を飛んでいるものはジャンプをしないと攻撃できないし，逆に飛び道具などの攻撃を（ものによっては）ジャンプでかわすこともできます。

シナリオ面での難易度はかなり甘口で，マップも単純なのですが，アクション的にはかなりキビシイです。前作以上の手さばきが必要になります。

見かけは「イース」などに似ていますが，戦闘システムは前述のVRシステムのせいでまるっきり違います。

まず，半キャラずらしはやめましょう。たいていの場合，自分のほうが大きなダメージをくらいます。それから，剣を出して突っ込んでいくのではなくて，敵が来るまでしばし待ち，剣が当たりそうなところで振ります。むやみにアタックしていくと，あっという間に死んじゃいます。

これが面倒なら，剣に魔法をかければフォースショット（弾が出る）を打つことができます。MP（マジックポイント）が減るので回数は有限なのですが，飛び道具なので，離れたところから攻撃できます。このゲームではモンスターが比較的お金持ちで，お金に困ることはあんまりないので，さっさと魔法屋さんで剣に魔法をかけてもらったほうが賢明です。というか，最初の段階でも，装備一式に魔法がかかってないと相手にならないくらい，ウェービスを代表する剣士ラトクさんは弱いんですよ。スライムさんてば強いのね。

私は実生活でも，安いものをたくさん買うより，なにがなんでもいいものを買う主義なので，このテのゲームでも最初からなるべくいいやつを買ってしまいます。ああ，もうちょっとで買えるわ，とお金を溜めていくのが至上の喜びなのですが，このゲームではそんなことはできません。

なぜって，ものを買うにはそれ相応のレベルが必要なんです（前作もそうだったかな？）。いくらお金持ちでも，レベルが足りないと武器屋さんが「これは君には使いこなせないよ」と売ってくれません。買っておけば，冒険の途中でレベルアップしたときに装備変更できると思うんですが，なかなかおせっかいな武器屋さんなのです。

もっとも，この武器屋の親父は，ラトクの親父の親友だったらしいので，それで心配してくれるんでしょう。そういうことにしておきましょ。ただで最初のくたびれた装備を直してくれることだし。

ゲームをやってみましょ

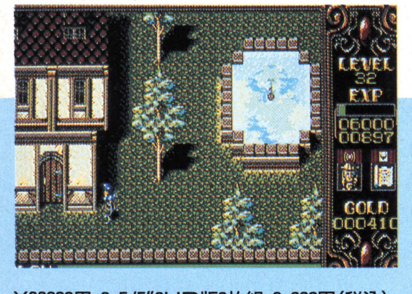
ゲームが始まると，まず森の中にいます。しばらくすると，ピクシーが飛んできて，女の子が縛られていると教えてくれます。

彼女の名前はシャナというそうで，森の種族の末裔だそうです。助けてあげると，お礼に町まで連れてってくれます。

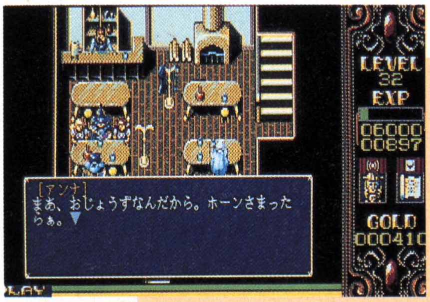
例によって例のごとく，町の中ではいろんな人と話しまくります。意外とみんな物知り。勇者ラトクが最初は倒せないくらい強いスライムや，大きなカエルさんが町の周辺にはいっぱいいるのに，みんなどうやって情報を仕入れるんでしょ～ねえ。

さて，まずは教会に行ってみましょう。ここではカルマの除去と，MPの回復をやってくれます。しかし，献上金として，しっかり100GPとられます。町のために働いてるんだから，ただにしてくれてもいいのに。

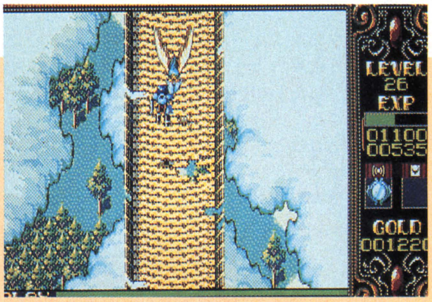
そうそう，このカルマってのがクセモノ。



X68000用 3.5/5"2HD版2枚組 6,800円（税込）
ブラザー工業（TAKERU） ☎052(824)2493



酒場の中でも話しまくろう



ハーピーのいる橋では落ちないように

モンスターを倒すと必ず上がり，一定以上溜まるとEXP（経験値）が上がらなくなります。１カ所で留まって経験値稼ぎをしてると，結局MPがなくなりカルマが溜まるので，このゲームの基本はレベルアップに適当な敵のいる場所と教会との往復に専念することになります。

さて，教会に行くと，森の魔物を倒してほしいと頼まれます。YES/NOの選択はあるのですが，NOにするとピクシーはうるさいし，ゲームが進まないので，しようがないからやってあげましょう。ゲームが進んでいくと，みんななかなかお調子者で，「あなたには，○○を倒してもらいます。それがあなたの宿命なのです」とばかりに，いろいろ鬼退治に行かされてしまいます。

そうそう，このゲームの基本はYESですからね。NOの選択肢はないも同然です。だってピクシーがうるさいんだもの。

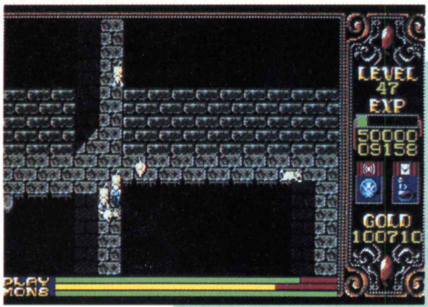
教会に行ったらシャナのところ（病院）に行って，偶像をもらってきましょう。これがないと森を自由に動き回れません。

森の中を動き回れるようになったら，木の上にあるおばあさんの家に行きましょう。なになに？　北の洞窟から鏡を持ってこい？　なんで私が……。いえいえ，こっちの話です。喜んで行かせてもらいます，って感じで鏡を取りにいかされます。

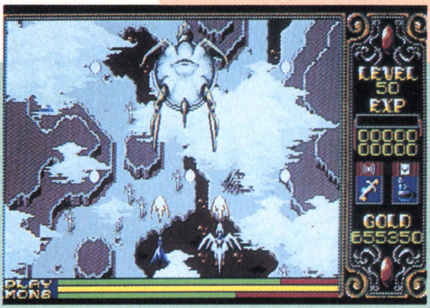
北の洞窟の前は，経験値の稼ぎどころですので，ここでレベルを32くらいまで上げましょう。そうすれば，新しい装備に新調できます。ここからはファルシオンソードが使えないと結構きついのです。

鏡をおばあさんのところに持っていくと，なぜかシャナがいます。どうもここはシャナのお家みたい。今度は空にかかる橋を通って，北の森に行けっていわれます。なんで私が……。いえいえ，こっちの話です，とあきらめて従いましょう。

北の森に行く途中には，ハーピーちゃんがいます。ハーピーは飛んでいるので，ジャンプしないと攻撃できません。また，ハーピーのいる橋はたてつけが悪く，ところどころに穴が開いているので，落ちないように。落ちるとすぐ死んじゃいます。



ねずみにかじられそう



ようやくやってきたシューティング面

そうして着いた先が北の森です。洞窟が２つほどあるので，制覇してください。ダンジョンのマップは簡単だし，敵はフォースショットでびしばしやっつければいいので，ここで経験値を稼ぎましょう。レベル34，35ぐらいまで上げておけば，ボスと比較的楽に戦えます。

レベルアップがテーマ

クエストはどれをとっても簡単です。とにかく，このゲームでは何をするにもまずはレベルと，ボスを倒すことができるような達人級の指さばきが必要になります。

前作「サーク」のときから，こういう印象は強かったのですが，「サークII」はもっとすごいです。しかも前述したように，カルマは溜まるし，肉弾戦ではすぐに負けてしまうのでMPが必要という理由から，教会との往復を強いられます。

そして，前作同様に「サークII」もボスがかなり強いです。特に最後のボスは強すぎるうえにかなりランダム性があるので，運がよくないと倒せません。

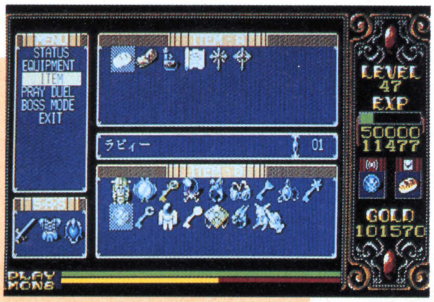
また，シューティング面もあるんですが，これもベタ移植でスクロールはカクカク，キャラクタは見にくい，目は疲れて涙が出ちゃうし。目が死ぬほど疲れたときにボスが出てきて，これもカタい。編集室でみんなでこの面をやっていたときに提案されたコツがなんと，「ここでポーズをかけて，目を休める！」なのですから(普通は無理。念のため)。

レベルアップの作業もそれほどいやなことではありません。しかし，シナリオ進行上必要なことをやれば，必然的にレベルは上がっているというのが自然ではないでしょうか。教会との往復の末にレベルアップ，レベルさえ上がれば簡単に解けてしまうクエスト，強すぎるボス。少なくとも私の視点で見れば，あまりバランスがいいとはいえないと思います。

「イース」がこのテのゲームの至高の作品とはいいません(X68000版は結構ボスキャラが強かったし)。しかし，もう一度やってみようという気が起きたり，実際に二度以上やってみた人もいると思います(「イースIII」はボスがカタすぎますが)。

ゲームシステムもちょっと難儀なところがあります。このゲームのアイテムは大きく分けて２種類あり，ひとつはヒールポーションのように意図的に使うアイテム，もうひとつはシナリオを進ませるうえで絶対必要なアイテムです。これらのアイテムはひとつしか装備できないので，後者のようなアイテムが必要な場所ではいちいち装備しなおしたりしなくてはならないのです。

最後のほうの砦に至っては，○○と鏡のアイテムつけ替えの応酬になってしまいます。このようなアイテムは装備するのではなく，持っているだけで有効にしたほうがいいと思うのは私だけでしょうか？



たくさんのアイテムを使い分ける

そっくりなのはいいことだ？

768×512モードの真ん中に，640×400の画面を表示しているため，画面がすごくちっちゃいです。

PC-88VA版では３枚だったのに，X68000版では２枚組になっていて不思議に思ったんですが，どうも1.44Mバイトフォーマットがかかってるみたいです。撮影の都合上，セーブが３カ所しかできないのには困ったので，ためしに2HDEでフォーマットしたディスクへとセーブ時に入れ替えたら（強制イジェクト）ちゃんとセーブできました。

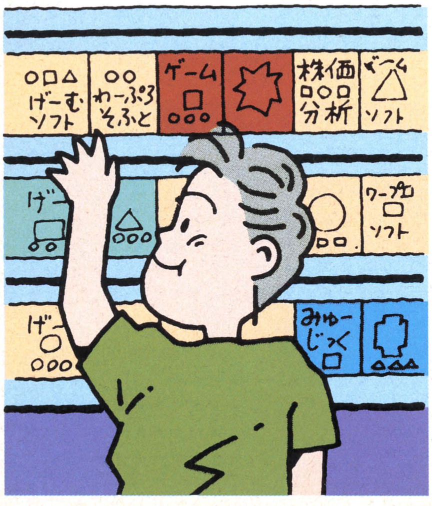
曲データのほうもほとんどベタ移植っぽいのですが，実はAD PCMを使ったパーカッションのアレンジがかかっています。そうそう，音源ドライバはZ-MUSICを使っているようですね。市販ソフト採用の第１弾かな？　あっ「ポニオン」があったか。

総合評価（0　5　10）

シナリオ難易度	★★★★
アクション難易度	★★★★★★★★★★
音楽	★★★★★★★
目が痛い	★★★★★★★★

AFTER REVIEW

今回はパッケージソフトではなく創刊10周年記念PRO-68Kに収録された，Oh!Xスタッフ謹製の「SION II」が登場。黄金週間PRO-68Kで発表された「SION」を遥かにうわ回る完成度で，多数反響があったようです。



SION II

▶SIONIIは面白いぞ！　スターウォーズとスターブレードを足してSIONで割ったようなゲームで，すごくいい。音楽もいい。でも，ちょっとランクがイージー。最後のボスのミサイル攻撃がスターブレードのボスの攻撃に似ているのは気のせいだろうか。

青木　学(17)秋田県

▶SIONIIにはまった。スターウォーズより面白いと思うぞ。しかし，キーボードでプレイするのはつらい。僕は3本指コントロールの人だから，どうしても上下の反応が遅れがちなのであった。

高橋　毅(21)埼玉県

▶SIONIIはよい。SIONIIIに期待してしまうのだけど，浜崎さん作る気あるのかな。

三浦　正義(19)神奈川県

▶SIONII，なかなかどうして素晴らしい出来ですね。これがたったの36円なんて((780－600)÷5）得した気分ですよ。

松永　正弘(22)京都府

▶SIONIIはすっごくよくできていますね。感動しました。SC-55を使用していますので，BGMがすごくいいです。3面に入ったときの“GATHER WAY!”がいちばんですね。

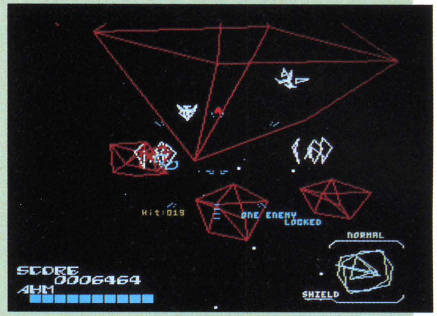
白沢　桂一(24)千葉県

▶SIONIIはむちゃくちゃよかった。通路と基地内部，それにボスの部屋に入ったところが最高。

仲井　隆良(14)愛知県

▶オープニングデモは，スタークルーザーかスターブレードみたいで格好よかった。ゲームスタート時のデモは少しいらないと思った。ゲームははっきりいって最後の迷路はいらない。余分だ，蛇足だ。曲がるとき一瞬止まるし，避けられない（？）シャッターはあるし。あとは，まあまあだった。ただ，ディスク1枚も使っているくせに，SIONよりゲームが短いのはなぜなんだ。

阿部　裕康(19)東京都



▶前作のSIONのほうが難しかったですね。SIONIIはすぐ終わってしまいました。AHM連射でラストのボスがすぐ死んじゃうんだもの。しかし，記録されているハイスコアは高すぎるぞ。全然届かない。

宮城　義和(29)静岡県

▶SIONIIの障害物が動いているところが熱いです。だから画面がいつも真っ赤です。

菅谷　英明(25)兵庫県

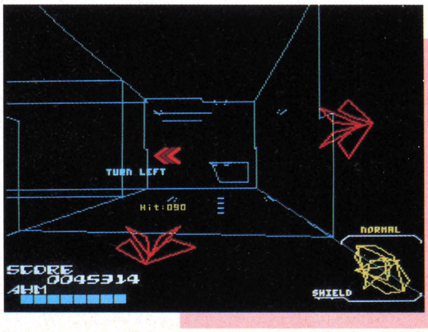
▶SIONII，感動のあまり涙が出た。カッコイイ，あまりにもカッコよすぎる。前作SIONのときはこんなふうだった。「これ，なかなか遊べるじゃん！　なんか得した気分」今回のは違う「もう何もいらない……」。最高です（M1対応もいつか）。

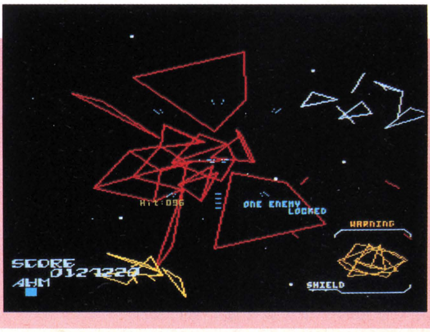
久保田　文彦(31)長野県

▶SIONIIは面白い。しかし，いまだにクリアできないでいる。

萩原　亮一(17)神奈川県

▶面白いけど，やっぱりスターウォーズをやったあとだと爽快感に欠けるなあ。ミサ





イルはいいけど数制限があるし。
長井　貴志(23)神奈川県

▶どうやったらあんな点数が出るんだ！どう頑張ってもEASYで14万点だぞ。これでAHMが10だなんていわれた日には……。でもヒットカウントだけなら勝ったぜ。ちなみにEASYで109，NORMALで103。やる気になればもっといけるのでは。
間宮　義晴(17)山形県

▶SIONII，ほとんど毎日プレイしています。久々に“クリアする楽しさ”を思い出させてくれるゲームでした。
加藤　真澄(？)京都府

▶ゲームは見せることではなく，魅せることであるとSIONIIから感じました。スピード感，ミュージックともに感動してしまいました。
伊藤　直樹(24)愛知県

▶だあ！　SIONIIは難しすぎるぞ。テストプレイヤーは何をしていたのだ。編集部の人間を基準に考えてはいけない。
鹿又　健(22)栃木県

▶SIONIIはとてもいい出来だ。MAGIC4.X，ZMUSIC.Xの性能も手伝って，SIONを遥かにうわ回る完成度に仕上がっていると思う。しかし，残念なことにゲーム内容にオリジナリティが全然感じられない。確かにプレイしていて面白いことは面白いのだが，どこかで見たようなシーンの連続には閉口せざるをえない。せっかく高い技術をもちながら，市販の人気ゲームのパクリに走ってしまっているのは，あまりに情けないことではないか。このゲームにオリジナリティという要素が加わっていれば，本当に素晴らしい作品になっていたと思うと残念でならない。

少しきついことを書いたが，SIONIIはもはや単なるサンプルゲームではなく，ひとつの完成されたゲームとして評価されてしかるべき作品だと思う。次回作に期待したい。
都築　二郎(20)岡山県

ちょっとアフターケア

さまざまな反響のあったSIONⅡですが，読者からの手厳しい意見もちらほら届いています。ゲームデザインに関することは，対処のしようがありませんので，隠し機能として対処できるもの，ちょっとした操作で対処できるものについて説明していきたいと思います。

まずは，いちばん多かった意見，

「ゲームが難しすぎる」

です。これは，“XF1”キーを押しながらゲームを立ち上げることによって，誰にでもクリアできるゲームになります。やり方は，オープニングが始まるまで“XF1”キーを押しっぱなしにして，コンフィグモードに移ってください。すると，見たこともない〝SPECIAL〟というメニューが追加されていると思います。ここで，以下の3つのモードを選択できます。

- NORMAL……通常のゲームモードにします
- AHM FREE……ホーミングミサイルが撃ち放題になります
- MUTEKI……無敵状態，つまり当たり判定なしでゲームを遊べます

ちなみに，無敵＆ホーミングミサイル撃ち放題はありません。どうしてもやりたければ，ソースリストを改造してくださいね。

それにしても結構露骨にわかるようにしておいたんですが，6月号のハガキには1通も発見したという報告がありませんでした（感づいている人はいた）。ま，隠し機能ですからいいのかもしれませんけどね。

次は，

「収録されているハイスコアが高すぎる」

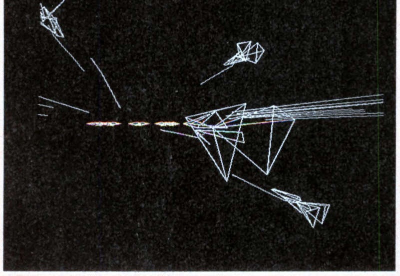
です。これの対処方法は簡単で，ハイスコアファイル「HI_SCORE.DAT」を削除するか，別のファイル名にリネームするだけです。このままゲームを立ち上げると〝FILE READ ERROR!〟の表示が出て止まりますが，無視してトリガボタンを押してください。すると起動は継続して行われますので，ハイスコア表示モードに移ってからデータのセーブを行えば，リセットされたデータを作成することができます。

ここでハイスコアを取るためのコツを説明します。まず，ゲームクリア時には，

シールドの残量×500
AHMの残量×1,000

のボーナス得点が加算されます。つまり，なるだけシールドを減らさず，AHMを使わないようにするのです。そして，ノーダメージクリアのときには特別に50,000点加算されます。さらに，ラスト面の最後に出てくる扉が1枚なんと10,000点！　全部で4枚出てきますから合計40,000点が稼ぎ出せます。実際にはビーム連射のAHM連射，かつタイミングが難しいため，かなりきついのですが不可能ではありません。

現在のハイスコアは，LIVE in'92でお馴染みの進藤氏が20万点台を叩き出しています。宇宙面では高得点の敵しか狙わず，ノーダメージ，扉4枚破壊するという徹底ぶり。でも，ハードランクでノーダメージクリアは達成していないそうなので，誰か挑戦してみませんか？　(M.H.)



どうなる!? SIONⅢ(仮)

発売中のソフト

★シュートレンジ　ビッツー
X68000用　5″2HD版3枚組　9,800円(税別)
★ポピュラスⅡ　イマジニア
X68000用　5″2HD版2枚組　12,800円(税別)
★ライジング・サン　ビクター音楽産業
X68000用　5″2HD版3枚組　9,800円(税別)
★ネクタリス　システムソフト
X68000用　5″2HD版　7,800円(税別)
★サークII　ブラザー工業(TAKERU)
X68000用　3.5/5″2HD版2枚組　6,800円(税別)
★チェイスH.Q.　ブラザー工業(TAKERU)
X68000用　3.5/5″2HD版2枚組　7,800円(税別)

新作情報

★ふしぎの海のナディア　ガイナックス
X68000用　3.5/5″2HD版　14,800円(税別)
★エトワールプリンセス　エグザクト
X68000用　5″2HD版　9,800円(税別)
★ヴェルスナーグ戦乱　ファミリーソフト
X68000用　3.5/5″2HD版　9,800円(税別)
★キャッスルズ　ビクター音楽産業
X68000用　5″2HD版　9,800円(税別)
★エアーマネージメント　光栄
X68000用　3.5/5″2HD版　11,800円(税別)
★バトル　ジー・エー・エム
X68000用　3.5/5″2HD版　12,800円(税別)
★沈黙の艦隊　ジー・エー・エム
X68000用　3.5/5″2HD版　12,800円(税別)
★オーバーテイク　ズーム
X68000用　5″2HD版　9,800円(税別)
★デスブレイド　SPS
X68000用　5″2HD版　価格未定
★餓狼伝説　ホームデータ
X68000用　5″2HD版　価格未定
★究極タイガー　KANEKO
X68000用　5″2HD版　価格未定
★バーンウェルト　グローディア
X68000用　5″2HD版　価格未定
★ファルディア　M.N.M Software
X68000用　5″2HD版　価格未定
★ドラゴンスレイヤー英雄伝説　SPS
X68000用　5″2HD版　価格未定
★鮫！　鮫！　鮫！　KANEKO
X68000用　5″2HD版　価格未定
★TATSUJIN　KANEKO
X68000用　5″2HD版　価格未定
★エアバスター　KANEKO
X68000用　5″2HD版　価格未定
★サバッシュⅡ　ポプコムソフト/グローディア
X68000用　5″2HD版　価格未定
★倉庫番リベンジ/ユーザー逆襲編　シンキングラビット
X68000用　5″2HD版　価格未定

概 要 紹 介

SX-WINDOW開発キット

Nakano Shuichi 中野 修一

X68000ユーザー待望のSX-WINDOW用開発ツールキットが発売されます。ここでは開発途中版からうかがえるツールの内容や操作性などについてまとめてみました。ウィンドウ上での開発環境はどのようになるのでしょうか。

編集部にSX-WINDOW開発キットの開発途中版が届きましたので概要をお伝えします。ただし，仕様はまだ正式に決定されたものではありませんから，発売直前版以降では変更されている可能性もあります。あらかじめご了承ください。

＊　＊　＊

さて，最近になってようやくSX-WINDOW対応のアプリケーションの話も聞こえてくるようになりましたが，これまでSX-WINDOWには正式な開発ツールというものがほとんどありませんでした。そもそも，SX-WINDOWが発表されたのが1990年春ですから，2年以上のあいだおあずけをくっていたことになります。

実は，これまで一般のソフトハウスに配布されていた資料というのは1991年1月号の謹賀新年PRO-68Kに入っていたものとなんら変わりません。それ以上の情報はほとんど公開されていませんし，ツールも存在しません。シャープ製のアプリケーションではアプリケーション1本ごとにマネージャやドライバが新設されたりしているのと比べると，初のSX-WINDOW対応ゲームとなったシムアースなどは非常に劣悪な環境で作られていたのがわかると思います。

SX-WINDOW自体がまだ安定していない時期もありましたし，暫定版の資料やツールにもたびたび不備が見受けられました。そのたびにSXプログラマはシステムを解析したり，試行錯誤を繰り返してきたわけです。

今回紹介する正式なSX-WINDOW開発キットの発表はSX-WINDOWユーザーすべてにとって福音であるといえるでしょう。

それでは開発キットの概要を順に解説していきましょう。

SXデバッガ

SX-WINDOW上で動作するソースコードデバッガです。これまではSX-WINDOW上のプログラムをSX-WINDOW上で開発することはほとんど不可能でした。特にデバッグ作業はターミナルデバッガを使用しないとデバッガが画面を破壊してしまうので大変です。また，バグ発生時に参照すべき情報は多いのですが，普通のデバッガではそれを一覧することはできません。こういうときこそ，マルチウィンドウ環境がほしくなるものです。

開発キットに付属するSDB.Xでは，これらの問題を一挙に解決しています。もちろんマルチウィンドウでさまざまな情報を参照できます。さらに，マルチウィンドウで同時に複数のターゲットプログラムをデバッグできます。が，そうするにはメモリがそれなりに必要になります。このデバッガを使用するにはメモリ4Mバイトでもちょっと心もとないという感じです。

機能や特徴は，C compiler Ver.2についてきたSCDをマルチウィンドウにして高機能にした感じといえばいいのでしょうか。

SX-WINDOW上で動いていますから，ダンプウィンドウとソースウィンドウ，レジスタウィンドウ，そして実行中のプログラムまでが同じ画面に表示されます。もちろん，SDB.X以外にエディタ.Xを同時に立ち上げてソースコードを書き換えたりできますし，ダンプウィンドウにはスタックの内容やワークエリアの内容がリアルタイムに表示できます。サンプルメイクを使えばコンパイル/アセンブルする部分までSX-WINDOW上で統合されます。

使い勝手はかなり良好な部類でしょう。デバッグプログラムのトレース実行などではモードの移行に少し戸惑いましたが（トレース実行中はほかのプログラムが止まる)，プログラムの性格上しかたのないことでしょう。

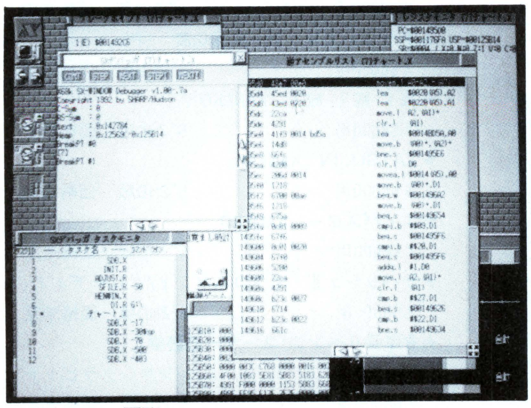
デバッグの手順を見てみましょう。

まず，ターゲットプログラムの指定です。これにはデバッグしたいプログラムのアイコンをタスクモニタに放り込みます。または，現在起動されているプログラム一覧から選んで指定してもかまいません。

ターゲットプログラムが指定されると，画面上に次々とウィンドウが開きます。

コンパイル時にデバッグ情報の付加を指定しておくと，自動的にソースプログラムをエディタウィンドウに表示します。

メモリダンプだけみても，単純ダンプからASCIIダンプ，浮動小数点数ダンプなど表示形式が多彩です。ブレイクポイントの設定やステップ実行などはSCDと同様にマウスひとつで簡単に行えます。ウィンドウアプリケーションの開発に限らず，X68000でもっとも優れたデバッグ環境というものがここに提示されています。


SXデバッガの画面

サンプルメイク

この開発キットのサンプルプログラムをコンパイルするためのMAKEです。MAKEの機能を果たすといっても，このプログラムはXCに付属のMAKE.Xをウィンドウ上から呼び出せるような環境を提供するもの

ですので，これ自体がMAKE機能を持っているわけではありません。もちろん，サンプルプログラム以外のプログラムで使えないようなこともありません。

とりあえず暫定版のSX-WINDOW版開発環境といったところでしょう。

基本的に開発するプログラムはひとつのディレクトリにまとめておいておきます。そして，サンプルメイクのウィンドウにターゲットのディレクトリアイコンを放り込むと，そこにあったMAKEFILEに従ってコンパイラなどを順次起動していきます。ソースのエディットからオブジェクトの生成まで，すべての工程をSX-WINDOWから実行できるようになったわけです。

サンプルメイクウィンドウはコンソールのように動作しますが，キー入力などは受け付けません。単にコンソールへのメッセージ出力をウィンドウ内に表示するだけです。もう一歩進めばコンソールが実現できるのですが……。

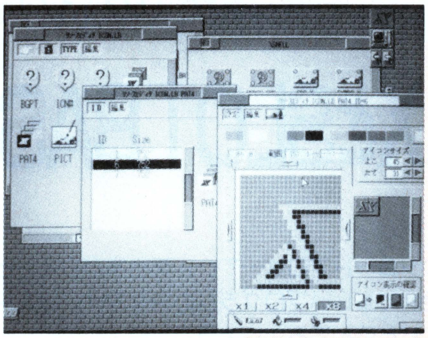
リソースエディタ

リソースエディタはSX-WINDOWプログラムがリソースとして使用するデータなどを編集するツールです。

基本的な使い方から解説しましょう。ウィンドウに＊.LBのアイコンを放り込むと登録されているリソースの一覧を表示します。

標準的なデータ構造ならば，それぞれをアイコン表示し，エディットすると専用のエディタが起動します。未知のものであれば？マークのアイコンが表示され内容を16進ダンプします。

WIND	ウィンドウ
DLOG	ダイアログ
DITL	ダイアログ内の配置
CTRL	コントロール
MENU	メニュー
TMPL	テンプレート
PICT	グラフィック
PAT4など	グラフィックパターン

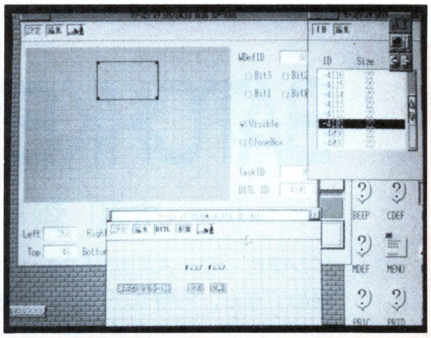

リソースエディタによるパターンの編集

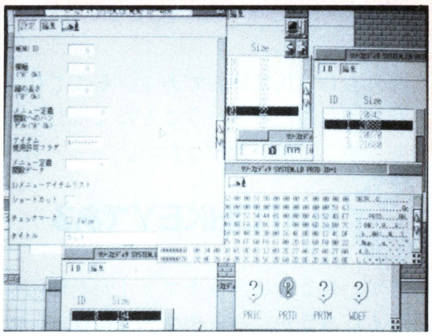
などのアイコンが用意されています。

ダイアログでは方眼紙状のウィンドウ上にダイアログを開く位置やダイアログの大きさを矩形で指示することができます。ダイアログ上のボタン類はまた別のウィンドウで配置などを設定することができます。PAT4などのグラフィックパターンでは「パターンエディタ.X」のようなものが起動します。

なお，現在アイコンは入っていませんでしたが，ほかにも設定項目をチェックリストや文字表示で表示するエディタが起動するものがあります。それ以外のビープ音やコードリソースなどはダンプでしか見られません。

しかし，サポートされているものに関しては実に簡単にエディットができます。必要な情報はわかりやすく表示され（それがどういう情報か把握することのほうが難しい），エディットできます。ボタンの大きさをマウスでちょいと変え，コントロールの種類を変えていくと画面の表示がすぐに変わっていきます。なかば当たり前のことですが，やはり対話性のよいソフトは使っていて楽しいものです。多少，全体の見通しがよくないもののひととおりの作業はこなせるツールです。


ダイアログだとこんな感じ


各種設定を変更できる

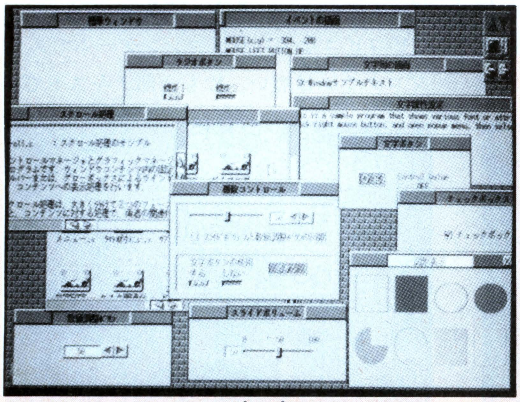
サンプルプログラム

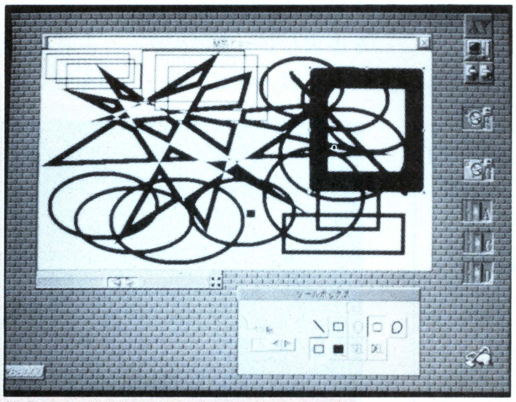
サンプルは基礎，応用，実用編と3つに分類されます。基礎編では，各マネージャの機能をそれぞれひととおり解説しています。応用編ではそれらを組み合わせた使い方，そして実用編ではゲームやテキストエディタ，グラフィックエディタ，ドローツールなどの簡易版が示されています。これらはいずれもC言語によるものです。

これらを参考に拡張すれば各種アプリケーション開発の労力はかなり軽減されることでしょう。もちろん，標準的なスケルトンプログラムも示されますので，まったく違うタイプのアプリケーションを作成する場合でも多少負担が少なくなります。

マニュアルでは，ようやくSX-WINDOWのインタフェイスガイドラインなども明示されるようです。まだ手元にマニュアルがありませんので，詳しいことはわかりませんが，マニュアルのデキ次第でこの開発キットの評価も大きく変わるでしょう。充実した内容のものを期待したいところです。

発売版を入手したらまた追加レポートを行う予定です。それでは皆さん，RAMを増設して発売に備えておきましょう。


各マネージャを使ったサンプルプログラム


簡易グラフィックツール

★(大)のショートプロぱーてぃ その37

BATCHKEY.XとSLOT.BAS

Yamato Satoshi 大和 哲

ピンチヒッター大和哲が送る（大）のショートプロぱーてぃ。雰囲気がちょっと硬めですが，内容はいつものとおり便利で楽しいプログラムの紹介です。リストは短いので，気分転換に軽い気持ちで打ち込んでみましょう。

illustration : T.Takahashi

今月も(で)氏は休みである。そこで私，大和哲が急遽代理ということになった。あらかじめご了承いただきたい。

BATCHKEYである

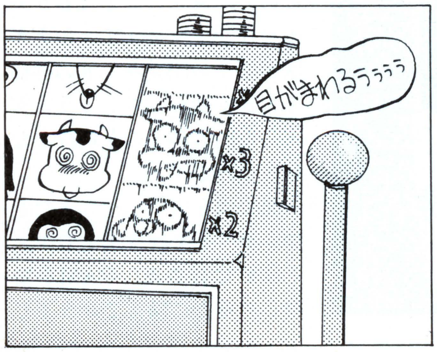
さて，皆さんはバッチファイルを使われているだろうか。バッチファイルは，一度使いたいコマンドの名前を連記しておくだけでよいという，便利，かつ，必要不可欠のものである。Human68kのシステムディスクにも入っているAUTOEXEC.BATなどもバッチファイルである。

しかしながら，このバッチファイルというものは，ただひたすらにコマンドを並べるものだけなら簡単に作れるが，会話型のプログラムを作るのは非常にむずかしいところがあった。なぜなら，バッチの文法にはキー入力に対する制御のためのコマンドが存在しないに等しかったのである。正確にいえば，キー入力を待つことはできたのだが，

作業を続けていいですか(y/n)？

のように，「yとnどちらが押されたのか知りたい」などというときには，手の打ちようがなかったのである。

そこで登場するのが今月のこのツールである。今月の１本目は奈良県の森芳生氏の作品でバッチファイルを便利にするためのコマンドBATCHKEY.Xである。



BATCHKEY.X for X68000

（要アセンブラ）

奈良県　森　芳生

このプログラムはバッチファイル中でのキー入力を可能にするプログラムである。プログラムリストはアセンブラのソースリストになっている。このプログラムを実行するためにはリストをBATCHKEY.Sという名前でエディタで打ち込んでから，アセンブラ，リンカを通してBATCHKEY.Xを作ること。

このBATCHKEYコマンドは，バッチファイル中のキー入力を行いたい場所で，

BATCHKEY　機能番号　メッセージ

と書くことでその機能を使うことができる。コマンドはバッチファイル中に書いたメッセージを表示し，機能番号に応じた処理をする。

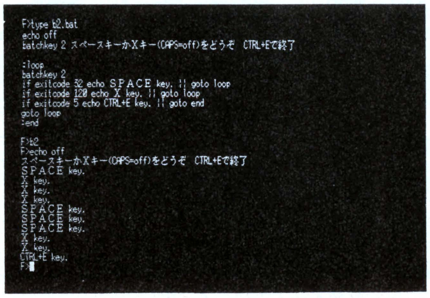
処理後，どのキーが入力されたかがエラーコードの形で返ってくるので，

IF～ERRORLEVEL
EXITCODE

などと組み合わせて使う。

機能番号は０～２の値を使い，機能番号０は，ＹキーまたはＮキーの入力要求を行う。それ以外のキーは，受けつけられない。Ｙキーが押されれば０，Ｎキーが押されれば１のエラーコードをBATCHKEYコマンドが返す。機能番号１は，０～９キーの入力を要求し０～９のエラーコードを，機能番号２では，任意のキー入力を行い押されたキーのキャラクタコードをエラーコードとして返す。

機能番号を指定しなかった場合，あるいは“？”を指定した場合には簡単なヘルプが表示される。また，メッセージは省略することができる。機能番号とメッセージの間は，必ずスペースを入れる。詰めて書くとプロンプトメッセージが，正しく表示されないので注意を要する。



BATCHKEY.S

初心者向けスロットゲーム

続いて，２本目のプログラムは石川県佐渡詩郎氏作によるX-BASIC用ピコピコゲームプログラム，SLOT.BASである。

SLOT.BAS for X68000

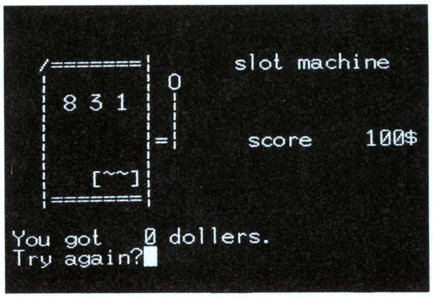
（X-BASIC）

石川県　佐渡詩郎

ファイル名どおり，内容はスロットマシンのゲームである。

リストを打ち込んでrunするとスロットマシンの絵が描かれ，スロットが回る。ここで1，2，3のキーを押すとそれぞれに対応するスロットが止まる。スロットが止まるとスロットに出た目に応じて賞金が支払われる。そして，もう１回やるか聞いてくるので，ここでやめるか，あるいはその時点でもう金がなくなってしまっていた場合，ゲームオーバー。そういうゲームだ。

ピコピコなスロットマシンゲームである。プログラムは短い。絵も全部テキストで描



SLOT.BAS

き……とにかくプログラムを短くしよう，という試みで作られたプログラムである。

投稿原稿いわく「どんな人にでも簡単に打ち込んでもらえるようにしたかった」。なるほど。しかし，それならば，人に打ち込ませるつもりならば，もう少し魅力を感じさせる工夫が必要ではなかったか？

スロットマシンはプログラミングの入門としてはわりとありふれた設定である。しかもこのプログラムは，画面もオールキャラクタである。ありがちで絵がきれいでない。残念ながらそれほどの工夫もない。

むろん，「どうせ出すなら打ちこんでもらえるプログラムを」と思った佐渡氏の気持ちはよくわかるつもりだ。しかし，それだからこそ人を引きつけるもう一歩の努力がほしかった，と思うのである。基本はできているのだから，ちょっとひねったアイデアを盛り込むように努力してみよう。

文句をいってしまったようだが佐渡氏にはこれからも頑張ってほしい。

それでは(で)氏の早期復活を願って，今月のショートプロぱーてぃを終わるのである。また来月。

リスト1 BATCHKEY.S

```
     1: *         バッチファイル用キー入力要求プログラム batchkey.
s
     2: *                              92/06/22  森ジャストミー
ト芳生
     3:
     4: *         .include        DOSCALL.MAC       *← このファイル
を
     5:                                             *   お持ちの方は
     6:
     7: DOS       macro   callname                  *↑
     8:           .dc.w   callname                  *|
     9:           endm                              *|この範囲は入
力
    10:                                             *|しなくて結構
です
    11: _EXIT     equ     $ff00                     *|  (4行目の'*'
を
    12: _EXIT2    equ     $ff4c                     *|   とって下さ
い)
    13: _GETC     equ     $ff08                     *|
    14: _PRINT    equ     $ff09                     *↓
    15:
    16:           .text
    17:
    18:           moveq.l #0,d0
    19:           addq.l  #1,a2
    20:           cmpi.b  #'0',(a2)
    21:           beq     in0
    22:           cmpi.b  #'1',(a2)
    23:           beq     in1
    24:           cmpi.b  #'2',(a2)
    25:           beq     in2
    26:           cmpi.b  #'?',(a2)
    27:           beq     usage
    28:           cmpi.b  #0,(a2)
    29:           beq     usage
    30:           pea.l   errmes
    31:           DOS     _PRINT
    32:           addq.l  #4,sp
    33:           DOS     _EXIT
    34: *-------------
    35: in0:
    36:           addq.l  #1,a2                     *↑
    37:           cmpi.b  #0,(a2)+                  *|
    38:           beq     lp0                       *|この部分は後
にも
    39:           move.l  a2,-(sp)                  *|出てきます
    40:           DOS     _PRINT                    *|
    41:           addq.l  #4,sp                     *↓
    42: lp0:
    43:           DOS     _GETC
    44:           cmpi.b  #'y',d0
    45:           beq     yes
    46:           cmpi.b  #'Y',d0
    47:           beq     yes
    48:           cmpi.b  #'n',d0
    49:           beq     no
    50:           cmpi.b  #'N',d0
    51:           beq     no
    52:           bra     lp0
    53: yes:
    54:           DOS     _EXIT
    55: no:
    56:           move.w  #1,-(sp)
    57:           DOS     _EXIT2
    58: *-------------
    59: in1:
    60:           addq.l  #1,a2                     *↑ほらね
    61:           cmpi.b  #0,(a2)+                  *|
    62:           beq     lp1      *←---------------+この行だけ
    63:           move.l  a2,-(sp)                  *|違います
    64:           DOS     _PRINT                    *|
    65:           addq.l  #4,sp                     *↓
    66: lp1:
    67:           DOS     _GETC
    68:           cmpi.b  #'0'-1,d0
    69:           bls     lp1
    70:           cmpi.b  #'9'+1,d0
    71:           bcc     lp1
    72:           sub.b   #48,d0
    73:           move.w  d0,-(sp)
    74:           DOS     _EXIT2
    75: *-------------
    76: in2:
    77:           addq.l  #1,a2                     *↑またもやでご
ざーる
    78:           cmpi.b  #0,(a2)+                  *|
    79:           beq     lp2      *←---------------+この行だけ
    80:           move.l  a2,-(sp)                  *|違います
    81:           DOS     _PRINT                    *|
    82:           addq.l  #4,sp                     *↓
    83: lp2:
    84:           DOS     _GETC
    85:           move.w  d0,-(sp)
    86:           DOS     _EXIT2
    87: *-------------
    88: usage:
    89:           pea.l   umes
    90:           DOS     _PRINT
    91:           addq.l  #4,sp
    92:           DOS     _EXIT
    93: *-------------
    94:           .data
    95:           .even
    96:
    97: umes:
    98:           .dc.b   '使用法 : BATCHKEY 機能番号 [プロンプト
メッセージ]',$0d,$0a
    99:           .dc.b   '         機能番号 0=Y 又は Nの要
求',$0d,$0a
   100:           .dc.b   '                 1=0～9キーの要求
',$0d,$0a
   101:           .dc.b   '                 2=任意のキーの要求
',$0d,$0a
   102:           .dc.b   '戻り値 : 機能番号=0の時',$0d,$0a
   103:           .dc.b   '           0=Yキーが押された  1=N
キーが押された',$0d,$0a
   104:           .dc.b   '         機能番号=1の時',$0d,$0a
   105:           .dc.b   '           0～9      ',$0d,$0a
   106:           .dc.b   '         機能番号=2の時',$0d,$0a
   107:           .dc.b   '           押されたキーのキャラクタコー
ド',$0d,$0a,0
   108: errmes:
   109:           .dc.b   '機能番号の指定が違います',$0d,$0a,0
   110:
   111:           .end
```

リスト2 SLOT.BAS

```
  5 screen 0,0,1,0
 10 str i$
 20 dim a(3),s(3)
 30 sc=105
 40 repeat
 50 sc=sc-5
 60 cls
 70 locate 0,3
 80 print "  /=======|       slot machine
 90 print "  |       | O
100 print "  | 0 0 0 | |
110 print "  |       | |
120 print using "  |       |=|      score ######$";sc
130 print "  |       |
140 print "  |    [‾‾]|
150 print "  |=======|
160 print "
170 s(1)=1:s(2)=1:s(3)=1
180 while s(1)+s(2)+s(3)
190   for t=1 to 3
200     if s(t) then a(t)=int(rnd()*10#)
210     locate t*2+1,5:print a(t)
220   next
230   i$=inkey$(0)
240   if i$="1" then s(1)=0
250   if i$="2" then s(2)=0
260   if i$="3" then s(3)=0
270 endwhile
280 if a(3)=a(1) then ds=10 else ds=0
290 if a(1)=a(2) then ds=ds+10
300 if a(2)=a(3) then ds=ds+10
310 if a(1)=a(2) and a(2)=a(3) then ds=ds+20:if a(1)=7 then ds
=ds+50
320 locate 0,12:print using "You got ### dollers.";ds
330 while ds
340   sc=sc+1:ds=ds-1
350   locate 18,7:print using "score ######$";sc
360 endwhile
370 locate 0,13
380 print "Try again?";
390 until inkey$<>chr$(13) or sc=0
400 cls
410 if sc=0 then print "Sorry,";
420 print "You have";sc;"dollers."
430 print
440 print "Thank you for playing."
```

打倒TORNADOへの第一歩(中編)

プロジェクトチームDōGA
かまた ゆたか

前回に続いて，お試しシステムのデータを利用して，F1のCGAを数カット制作してみます。できたカットをつなげてアニメーションすればちょっとした作品に，……なるかな？

はじめに

DōGAの恒例行事としては，CGAコンテストより歴史がある“CGA強化キャンプ”が8月8，9日に行われました。このキャンプは毎年鈴鹿山系の川辺にテントを張り，プログラム開発に必要な腕力を鍛えるために川で泳ぎ，バグを見つける検索能力を強化するために薪を探すという，どこから見ても遊んでいるだけという内容です。

しかし，今年は台風10号の上陸と重なって，テントは荷物ごと飛ばされるわ，川は土石流と化すわで，もうたいへん。暴風雨の中，“朝が来れば，きっと活路が開ける！”と祈りつづけたひと晩は非常に長く感じました(もうほとんど遭難者)。

台風情報を見るために持っていった携帯テレビ。なんで巨人－中日戦しか映らへんねん！

さて前回は，芸術祭でグランプリを受賞した文月さんの“誰にでもTORNADOは作れる”という言葉が，ただの謙遜か，真実かを確かめるため，実際に皆さんにCGAを作ってもらうという話でした。まだ，1回目ということで，物体もひとつで視点が動くだけというものでしたが，皆さんはちゃんと実行できたでしょうか。

引き続いて，「TORNADO」に対抗するF1のCGAですが，今回はちゃんと背景もつけ，物体も動かしてみます。

まずは準備

同じ内容を何度も書くわけにはいきませんので，先月号をちゃんと読んで，実行しているという前提で話を進めさせていただきます。しかし，先月号で使ったワークディスクで作業を続けていると，ディスクがいっぱいになるので，もう1枚新しいディスクを用意しましょう。

新しいディスクには，

TYRL.SUF	TYRL.ATR	JIMEN.SUF
JIMEN.ATR	F1A.FSC	

の5つのファイルを入れておいてください。

コピーの方法などがわからない方は，CGAシステムのマニュアルの「CGA大学/コンピュータ基礎概論」を勉強してください。

前回は，第1回ということで，操作手順，操作結果をすべて羅列していきました。しかしあの方法は，いくらなんでも誌面を食いすぎますので，前回と同じところなどはどんどん省略していきます。どうしてもわからないところ，うまくいかないところなどありましたら，当チームの「補習」係までお便りください。

背景をつける

前回制作した視点だけが移動するカットには，背景がありませんでした。これではちょっと寂しいので，今回は前回のカットに背景をつけることから始めましょう。

背景としてはお試しシステムの背景用地形データ「JIMEN」を利用し，さらに空の部分を水色にします。

○FFEを起動する

操作：コマンドラインから，FFE，リターンと入力する

解説：マニュアルの「CGA大学/PES基礎実習」の単位をちゃんと修得している人は，PESから起動してもらっても結構です。その場合，ワークディスクのドライブをカレントにしておいてから起動してください。そうでないと，先月号の柚姫のようになってしまいます。コマンドラインから起動しようとして，「コマンドまたはファイルがありません」などのエラーとなった場合は，先月号を参考にして，起動しなおしてください。

○モーションデータを読み込む

操作：1)　メッセージパネルの「6)　ファイル」を左クリックする

2)　「LOAD」を左クリックする

3)　「F1A.FSC」と表示されているところを左クリックする

→「データ読み込み中」などが表示される

解説：前回制作したモーションデータ「F1A.FSC」をロードしています。「F1A.FSC」で使用した形状ファイル「TYRL.FSC」を同じディレクトリにあらかじめ入れておいてください。

○「JIMEN.SUF」を設定する

概要：「JIMEN.SUF」を追加し，X，Y，Zの各方向に

10倍拡大して，さらにZ方向に「−70」移動させて，決定します。

操作：1）「4）物体設定」を左クリックする

2）「1）追加」を左クリックする

3）「JIMEN.SUF」と表示されているところを左クリックする

→「データ読み込み中」などが表示されたあと，平面図，側面図に野山の地形データが表示される（図1）

4）「拡大X： 1.00」と表示されているところの「1.00」を左クリックする

→「1.00」が反転し，入力状態になる

5）キーボードから，10，リターンと入力する

→反転がYの値に移動し，Y方向の拡大率の入力状態になる

6）キーボードから，10，リターンと入力する

→反転がZの値に移動し，Z方向の拡大率の入力状態になる。「JIMEN.SUF」が10倍拡大される（図2）

7）位置の「Z：0」と表示されているところの「0」を左クリックする

→Z座標入力状態になる

8）キーボードから−70，リターンを入力する

→地面とタイヤがちょうど接するようになる（図3）

9）「決定」を左クリックする

10）物体設定の「4）終了」を左クリックする

解説：前回「TYRL.SUF」を置いたように，「JIMEN.SUF」を設定しています。ただ，そのまま置くと丘と車が同じぐらいの大きさになってしまうので，「JIMEN.SUF」を10倍拡大する必要があります。図2を横から見て地面が2本の線になっている（つまり地面が2枚ある）のは，「JIMEN.SUF」を制作したときの手抜きのせいで，あまり気にしないでください。上のほうの線（Z＝0）だけを考えていただければ結構です。

もし，この「JIMEN.SUF」をZ方向に70下げないと，タイヤが地面にめり込んでしまいます。地面を下げずに，車を上げるという方法もありますが，背景はこの「JIMEN.SUF」だけで一度設定すればよいのに対して，「TYRL.SUF」はこれから何度も動かすので，地面を下げるほうが簡単でしょう。

だいたい，車，木，建物などの地面に置くものは，Z＝0の位置に置いたとき，ちゃんと地面と接するように作るべきです。「TYRL.SUF」がなぜこのように中途半端な位置に作られているのか作者の砂川君に聞いたところ“特に意味はない”ということ（要するにミス）です。

○背景を水色に設定する

概要：FFEの背景設定の機能で，何も物体がないところ（空の部分）に色をつけます。

操作：1）「1）背景設定」を左クリックする

2）「2）べたぬり」を左クリックする

→Rの値の入力状態になる

3）0.2，リターン

4）0.4，リターン

5）0.8，リターン

→背景色がRGB＝(0.2 0.4 0.8)に設定される

6）「決定」を左クリックする

解説：RGBの値は，このとおりにする必要はありません。Bの値が大きければそれなりに空色になるでしょう。

○セーブ・終了

操作：1）「6）ファイル」を左クリック

2）「SAVE」を左クリック

3）「フレームソース」を左クリック

4）出力ファイル名“X10A”，リターンをキーボードで入力

5）「7）終了」を左クリック

解説：前回と同様ですから，特に解説することはありません。出力ファイル名の「X10A」は，Oh!X10月号のカットAという意味です。

○作画・アニメーション

操作：1）コマンドラインから“AUTO X10A.FSC / A2 /G/D”を実行する

2）作画後，「動画 X10A について，出力ファイル名は？」と表示されるが，無視してリターンを押す

解説：PESを使ってもらってももちろん結構です。しか

図1 「JIMEN.SUF」を読んだ直後

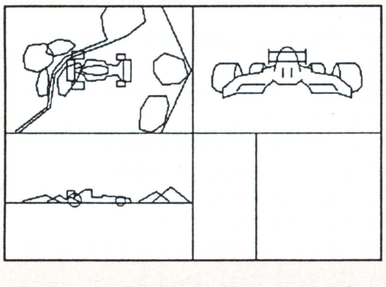


図2 「JIMEN.SUF」を10倍に拡大

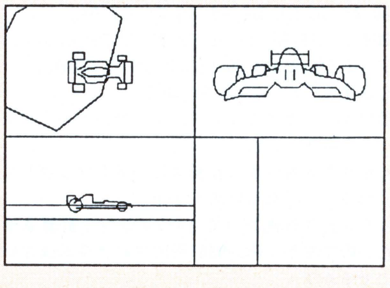
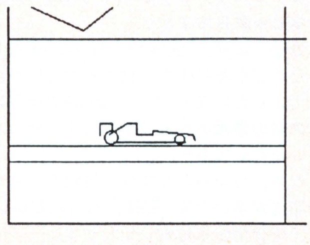


図3 タイヤが地面に接する



しPESのAUTOのオプションには，なぜか“/D”がありません（単なるミス）ので，あとでCRDを実行してやる必要があります。マニュアルの「CGA大学／制作環境システム文法」(P.285)を参考にして，PES.DEF を書き換えておくのもよいでしょう。

作画計算には相当時間がかかりますので，その間にコラムやほかの記事を読んでください。前回のアニメーションと比べると，背景がついただけですが，イメージはかなり変わったでしょう

物体を動かす

では，いよいよ「TYRL.SUF」を動かしてみましょう。「X10A.FSC」につながるカット，「X10B.FSC」を作ります。

前回，視点だけを動かすカットを制作しましたが，物体を動かすときもだいたい同じで，フレームナンバーを変更したあと，位置を変更してやります。視点と違うところは，視点は必ずひとつなのに対して，物体は通常複数存在するため，どの物体を動かすのか選択するという操作が増えます。

○1フレーム目の設定

概要：「X10A.FSC」の20フレーム目とまったく同じように，「X10B.FSC」の1フレーム目を設定します。すでに解説したので，詳しい操作は省略します。

操作：1)　FFEを起動する

2)　背景を水色に設定する

前述の「背景をつける」の“4)”と同様。「背景設定」の「べたぬり」で，RGB＝(0.2 0.4 0.8)

3)　「JIMEN.SUF」を設定する

前述の「背景をつける」の“3)”と同様。「物体設定」の「追加」のあと，各軸方向に10倍して，Z軸方向に“－70”移動する

4)　「TYRL.SUF」を設定する

先月号の「物体設定」と同様。原点にそのまま置く

5)　視点を変更する

先月号の「視点の変更」と同様。視点の位置は，(－400,800,600)，注目点は変更せず，(0,0,0)のまま

○フレームナンバーの変更

概要：操作自体は，先月号の「フレームNo.の変更」と同じですが，今回のフレームナンバーは5を設定します。つまりたった5フレームのアニメーションです。

操作：「5)　フレームNo設定」を選択し，フレームナンバー「5」をキー入力して，リターンで決定する。

読者によるほっとけないほっとこらむ

うさ子：今月は，夏休みに遊びに来た平等電子（たいらでんこ）ちゃんと一緒にお手紙を紹介させていただきます。

電子：ご紹介にあずかりました電脳倶楽部でお馴染みの新聞配達少女の電子です。うさ子おねえさまともどもよろしくお願いします。

うさ子：では，今回も先月に引き続き，DōGA CGAシステムver.2.50マニュアル申し込み書の自由記入欄に寄せられた多数のご意見，ご感想のなかから，ホットな生の声をご紹介しますね。

＜Aさん＞法人化なら，税金などを考えても，宗教法人がいいのでは？　信仰対象としては，シャープに金メッキのX68000を作ってもらっては……。では，頑張ってください。

うさ子：アドバイスありがとうございます。DōGAの薄暗いプロジェクトルームに，さんぜんと光り輝くマシンがあるのを想像すると，すごく不気味で，アヤシイ雰囲気になりそう。

電子：あっ，おねえさま。こっちには，学校法人がよいってお便りがありますよ。その場合，通信教育になるのかしらん。

＜Bさん＞今回のカンパも，お金以外も受け付けているのでしょうか？　売れ残りの漫画の単行本じゃ駄目ですか？

うさ子：以前，「究極超人あ～る」全巻をいただいたことがあります（島さんがたいへんよろこんでました）。でも，すでにDōGAの部屋にはかなりの数の漫画本がありますので，せっかく送っていただいても……。

なお，カンパとしては，食料も（食べざかりの若者が多いので。私はにんじんがいいな），そして肉体労働も歓迎いたします。

電子：でも，なまものの場合は，コックさんとウェイトレスをつけてね。

遊び人：僕は，お嫁さんがいい。

マリオ：カップラーメンがいい。

＜Cさん＞感謝の気持ちを込めて，カンパとして，お酒を送ります。ここ秋田の三大特産品は，米・酒・秋田美人です。本当は「秋田美人」を送ろうとしたのですが，クール宅急便では送れず，また，美人はすでに柚姫さんがいらっしゃるので，やめておきました。どうぞ，思い切り冷やしてお飲みください。

うさ子：どうもありがとうございました。でも，スタッフには「秋田美人」を送ってほしかったという人が多かったです。クール宅急便でなくても東北新幹線があるじゃありませんか。また次の機会には，ぜひ，ご検討ください。

遊び人：僕は「秋田美人」のお嫁さんがいい。

文月：よし，「秋田美人」は俺が受け取りにいく。

電子：あれっ？　なぜ，文月さんがここに。

文月：俺も遊びに来てたんだよ。

電子：じゃあ，帰りは乗せていってください。

＜Dさん＞「BA CB DF B0 A0 BB DE BE C2 A0 32 CC DE D2 A0 C0 C9 D1 A1」

電子：「B6 B2 C4 DE B8 20 C3 DE B7 C0 20 BC DE CC DE DD B6 DE 20 B6 C5 BC B2」

うさ子：「0；GD(4I PEB4PL」

遊び人：「カラミニノチノナ　ランラモイトチミミ　キチ　ニニ」

＜Eさん＞もう少し作画スピードをアップしてくれると，うれしいのですが。

＜Fさん＞作画だけでも486マシンで計算させたいのですが，PC-9801用のプログラムはないのですか？

＜Gさん＞せっかくのV70ボードに対応する計画はないのですか？

電子：う～ん。X68000の場合，とりあえずクロックアップすると，確実に速くなりますが。でも，責任はいっさいとれません。

うさ子：PC-9801，FM TOWNSのREND，FFなどはすでに開発されています。タケルなどでまとめて発表する予定ですので，いましばらくお待ちください。

かまた：V70ボードにも対応させてみたんですけど，期待したほどのスピードは出ませんでした。もっとも，これはボードのせいというよりコンパイラの問題で，まだ改善の余地はありますので，発表についてはもう少し検討させてください。

Taka2：ODPをつけたエプソンのPC-486GRだと，なんと約40倍に。

かまた：作画スピードのベンチマークテストについては，次回にでも報告します。

＜Hさん＞暑さに負けずに，がんばってください。

電子：スタッフは大丈夫ですが，24MHz化したX68000 XVIが……。

うさ子：扇風機による強制冷却にも限度がありますよね。

マリオ：シャープ様。クーラーありがとうございました。

文月：お～い，電子。そろそろ出発するぞ～。

電子：え～！　文月さん，待って待って～。ということで，うさ子おねえさま，おじゃましてすいません。それでは！

うさ子：なんか，今月はにぎやかでしたね。電子ちゃん，またお会いしましょう。

解説：詳しくは，先月号をご覧ください。「決定」だけがマウスの左クリックで指定できない（三保君はやっとバグと認め，バージョンアップをしています）ので注意してください。

○物体の選択

操作：1)　「4)　物体設定」を左クリック

2)　「2)　変更」を左クリック

→選択モードに入る

3)　「選択」を左クリック

→「物体名」の横に「tyrl」と表示されます

4)　「決定」を左クリック

→車が黄色で表示され，物体追加の位置設定と同じ画面になる

解説：選択モードに入ったとき，本当はX，Y，Zの値を入れたあとで「選択」をすることで，その座標にいちばん近い物体を選択するというのが正しい使い方です。しかし，この座標入力がマウスでできないため，かえって面倒になってしまいます（三保君にはちゃんとバージョンアップするよう交渉しておきます。だけど三保君も卒論で忙しいですから……）。

「選択」をクリックすると，最も近いものから順番に次々に選択されていくので，目的の物体が選択されるまで「選択」をクリックするという使い方をします。選択された物体の位置には，白い小さな長方形が表示されますので，同じ名前の物体が複数あっても区別することができます。

○物体の位置・向き設定

概要：物体を選択すれば，あとは物体追加の位置設定と同じです。

操作：1)　図4のように，マウスカーソルを平面図（左上の図）の右のほうに持っていって，左クリック

→メッセージパネルにその座標が表示される

2)　1)を繰り返して，座標がおおよそ（800，−100，0）ぐらいにする

3)　「作画」を左クリックする

→完成予想図が表示される（図5）

4)　メッセージパネルの「回転」の「Z：0」の「0」を左クリック

→Z軸回りの回転の値の入力状態になる

5)　キーボードから，−10，リターンを入力する

→（図6）

6)　「決定」を左クリック

7)　物体設定の「4)　終了」を左クリック

解説：物体の位置設定は，前回のように各座標値をクリックしてキーボードから入力する方法もありますが，通常のモーションデザインでは，座標値はあらかじめわかってはいません。ですから，適当な位置をマウスで指定しては「作画」という操作を，自分の思いどおりの構図になるまで繰り返して，最後に「決定」します。

ただ，この方法だと座標値に端数が出て気持ち悪いので，マウスでだいたいのアタリをつけたあと，きりのいい数値をキーボードで入力するといった使い方が一般的です。

なお，回転角度については，マウスで入力できませんので，キーボードで設定します。

○セーブ・終了

操作：前述の「背景をつける」の“5)”と同様。「ファイル」の「SAVE」で「フレームソース」を選択し，出力ファイル名「X10B」をキー入力する。そして，「終了」でFFEを終わる。

○作画・アニメーション

操作：AUTO X10B.FSC /A2 /G /D

解説：指定するファイル名が異なるだけで，操作は上記の「背景をつける」の“6)”と同じです。確かに車が動きますが，たった5フレームで，アニメーションとしては面白くともなんともありません。しかし，このカットは，あとでつなげてアニメーションするときに，重要な役割を果たします。ちゃんと残しておいてください。

視点も物体も動かす

それでは，最後に視点も物体も動くカットを作ります。とはいっても，特別な操作があるわけではなく，視点を動かすカット，物体を動かすカットの両方の操作をするだけです。まあ，今月のまとめとしてやってください。

フレームナンバーは，1と10と40の3つを指定します。つまり，曲線的な動きも表現できます。しかし今回は，

図4　変更位置を指定

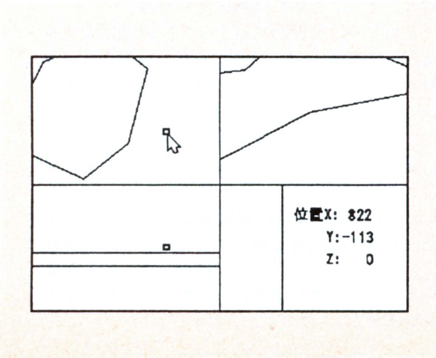


図5　完成予想図

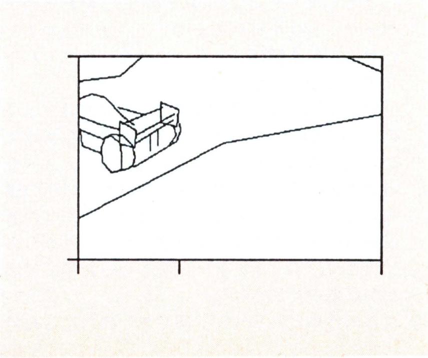
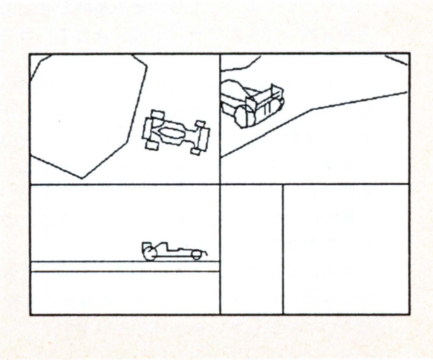


図6　物体回転



曲線的な動きをするのは視点だけで，「TYRL.SUF」自身はX軸方向に直進します（図7）。

○背景の設定

操作：1)　FFEを起動する

2)　背景を水色に設定する

前述の「背景をつける」の“4)”と同様。「背景設定」の「べたぬり」で，RGB＝(0.2 0.4 0.8)

3)　「JIMEN.SUF」を設定する

前述の「背景をつける」の“3)”と同様。「物体設定」の「追加」のあと，各軸方向に10倍して，Z軸方向に“−70”移動する

解説：もうすでに2度やった作業です。このカットを制作したあと，ご自分でいろいろ制作してみるつもりでしたら，ここでSAVEしておくべきでしょう。カットを作るたびにこの操作をせずにすみます。

SAVEの方法は，前回同様，「ファイル」の「SAVE」の「フレームソース」を選択して，適当な出力ファイル名（BACKなど）を入力してください。

○表示範囲の変更

概要：このカットでは，「TYRL.SUF」がかなりの距離を移動しますので，「TYRL.SUF」の設定の前に，画面左側の2面図の縮尺を変えておきます。

操作：1)　「＝」を押す

2)　「＝」を押す

3)　「＝」を押す

解説：「＝」を押すと，より広い範囲が表示されます。逆に「＋」を押すと現在表示されている中心付近が拡大されて表示されます。表示しなおすのには多少時間がかかりますが，「＝」を押しつづけているとキーバッファが溜まってどんどん書き換えられてしまうのでご注意ください。縮尺を変更すると，図8の位置に表示されているグリッドの値が変わります。グリッドとは，2面図の青のマス目の1マスの大きさを意味します。

○1フレーム目の設定

概要：「TYRL.SUF」と視点を設定します。視点設定で，注目点の座標も変更しますが，要領は視点の座標と同様に変更したい数値をクリックして，その数値が反転した状態でキーボードから入力してやります。リターンをすると反転がひとつ下の項目に移動します。

操作：1)　「TYRL.SUF」の設定

「物体設定」の「追加」で「TYRL.SUF」を指定する。位置座標を(−15000,6000,0)と指定し，「決

法人化への道

DōGAが法人化するかもしれないという話はすでに紹介しました。別に勝手に法人化すればいいといってしまえばそれまでですが，CGAシステムのユーザーは，DōGAの活動に賛同，参加の意思のある方々，つまり準スタッフみたいなものですから，できるだけオープンに話を進めていきたいと思っています。

そこで，このコラムでは，法人化にあたっての方針，問題点，苦労話などを紹介していきたいと思います。今回は，「DōGA法人化実行委員会」の第1回会合から抜粋します。

「マニュアル申し込み用紙の自由記入覧のところで，法人化について書いてある意見を見ると，ちょっと意外な結果だけど，ほぼ100%が法人化に賛成みたい」

「“アマチュアリズムを貫くべきだ”という意見が結構あると予想してたけど，なかったんとちゃうか」

「逆に“はっきり営利目的で活動すべき”という意見は結構あった」

「なんか，いままで“非営利”でがんばってきたのがバカみたい」

「いや，いままでこんな活動を展開できたのは，“非営利”だったからこそで，決してバカではないぞ」

「でも，多くのユーザーにとっては，“理想”とか“目的”とかいう理屈より，しっかりしたサポート体制，なるべくバグの少ないソフトのほうを望むってことでしょう。ちょっと寂しい意見だけど」

「確かに，我々の思いとユーザーの意識にはギャップがあるかもしれへんな」

「まあ，とにかく，法人化に関しては，読者の理解を得られたということで，具体的に実現化について検討していきたいと思います。う〜ん，このいい回し，PKOみたい」

「で，何から決めるの？」

「知らん。法人化って誰もしたことないもん」

「資本金はさ，1670万円とか，512万円とか，256万円にしよう」

「でも，株式会社は1000万円，有限会社は300万円以上やで」

「そういえば，かまたさんは財団法人がいいといってたけど，なんで？」

「理由は，2つある。まずひとつめは，株式会社や有限会社よりギャグとして笑える」

（この瞬間，数人のスタッフがズッコケ，数人のスタッフが深くうなずく）

「え〜，ギャグで法人化するの？」

「でも，そういったノリって大切だと思うよ」

「ちょっと説得力が……」

「2つめの理由は，財団法人は公益法人だから。自分たちの活動がいくら営利目的でないっていっても，株式会社や有限会社は営利目的の組織だから，そんなふうに見られるのがいや。名実一致させるためには，財団法人にしたほうがいいと思う」

「どうして，そんなに“非営利”にこだわるの？せっかく会社を辞めてまでして，法人化するんだから，お金儲けしたらいいやん。ソフトハウスのどこがいけないの」

「DōGAが目的とするところの“CGA文化”ってのは，高性能のCGのプログラムがひとつあればいいってもんじゃないねん。CGAコンテストや各地の上映会のような発表の場とか，Oh!Xの連載のように，作品の作り方のコツやみんなの意見を紹介する場とか，制作者やチームどうしの交流とか，ネット，データ集といった幅広い活動が必要やと思うわけや。それって，ソフトハウスの仕事とちゃうで。同じ思いのたくさんの人たちの行動があって初めて可能なこと。まあ，法人化したって，その活動は草の根運動が中心になるってこっちゃ」

「第一，法人化したって，儲かる保証なんかない。確実にお金儲けしたいんなら，一流会社でのうのうとしているほうがずっと楽」

「まあ，わからんでもないけど，財団法人はむずかしいで。法律に詳しい知人の話だと，認可はむずかしいし，最初の基金も相当な金額じゃないとあかんとか，公認会計士をつけて，厳しい管理を受けるとかあるらしいで。そんなん，いややな」

「“……らしいで”じゃ困るな。これから財団法人のことをきっちり調べて，何か問題点があるとはっきりしたら，その時点であきらめればいいやん」

「あきらめたときは，株式会社？　それとも有限会社？」

「有限会社がいいんとちゃう？　税制や設立の手続きが簡単だから。株式会社のほうが対外的な信用が大きいとか，“株式会社社長”を名乗りたいから無理して株式会社にするって話を聞くけど，そういうのには興味ないやろ」

「それじゃ，今回の結論として，財団法人化を目指して調査をし，だめだったら有限会社を検討するということで，そのセンでやってみましょう」

ということで，ご意見がある方は，当チーム「法人化への道」の係まで，どしどしお便りください。

P.S.“法人化の際には就職希望”のUさん。まだ，話はこの程度しか決まっていません。それに，法人化しても，そんなにたくさんの人を雇えるような大きな会社にはなれないと思います。すいません。

定」

2)　視点の設定

「視点設定」で視点の座標を（−15000,6300,30）と指定し，注目点の座標を（−14500,6000,30）として「決定」

解説：「TYRL.SUF」の位置設定は，上記の「物体を動かす」のようにマウスで座標を指定するのが普通ですが，今回は，座標値に厳密さが要求されるので，上記の値をキー入力するほうがよいでしょう。

○10フレーム目の設定

操作：1)　フレームナンバーを10にする

「フレームNo設定」を選択し，フレームナンバー“10”を入力して，リターンで決定

2)　「TYRL.SUF」を選択する

「物体設定」の「変更」で，「選択」を２回クリック。「物体名」が「tyrl」となったのを確認して「決定」

3)　「TYRL.SUF」の位置を変更する位置座標は(−10000,6000,0)。変化しているX座標の数値を入力するだけでよい

4)　視点を変更する

「視点設定」で視点の座標を（−11000,5700,30）と指定し，注目点の座標を（−9000,6000,30）として「決定」

解説：もうほとんど解説することはありません。物体選択のときには「選択」を２回クリックしますが，これは「JIMEN.SUF」のほうが原点に近いために，１回目の「選択」では，「JIMEN.SUF」が選ばれてしまうからなのです。

○40フレーム目の設定

操作：1)　フレームナンバーを40にする

「フレームNo.設定」を選択し，フレームナンバー「40」を入力して，リターンで決定

2)　「TYRL.SUF」を選択する

「物体設定」の「変更」で，「選択」を２回クリック。「物体名」が「tyrl」となったのを確認して「決定」

3)　「TYRL.SUF」の位置を変更する位置座標は(5000,6000,0)。変化しているX座標の数値を入力するだけでよい

4)　視点を変更する

「視点設定」で，視点の座標を（2000,5700,30）と指定し，注目点の座標を（5000,6000,30）として「決定」

5)　セーブ・終了

「ファイル」の「SAVE」で「フレームソース」を選択し，出力ファイル名「X10C」をキー入力する。そして，「終了」でFFEを終わる

○モーション確認

概要：５フレームならともかく，40フレームともなると，さすがに作画時間はかかります。そこで，作画計算に入る前に，モーションデータを確認しておきましょう。ワイヤーフレームで高速に計算する「WIREVIEW」というプログラムを使いますが，その前にフレームソースをフレームファイルに変換する「FF」を実行します。

操作：1)　FFを実行する

コマンドラインから，“FF X10C”と入力する

2)　WIREVIEWを実行する

コマンドラインから，“WIREVIEW TYRL.SUF JIMEN.SUF X10C.FRM /V”と入力する

3)　アニメーションを確認する

しばらく計算したあと，アニメーションが表示される。65ページの写真と見比べ，特に異常がないようなら，「ESC」を押してアニメーションを終了する

4)　作画・アニメーションコマンドラインから，“AUTO X10C.FSC　/A2 /G /D”と入力する。すぐに「すでに，X10C.FRM があります……」というメッセージが出るが，気にせず「１」を選択

図7　「TYRL.SUF」の動き

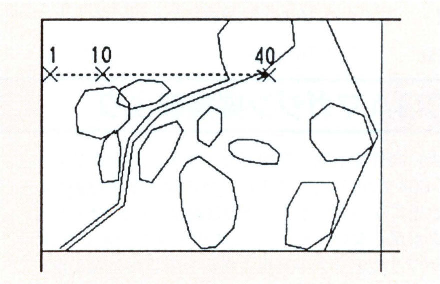


図8　グリッドの表示位置

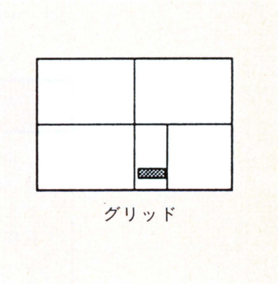


図9　１フレーム目の完成予想図

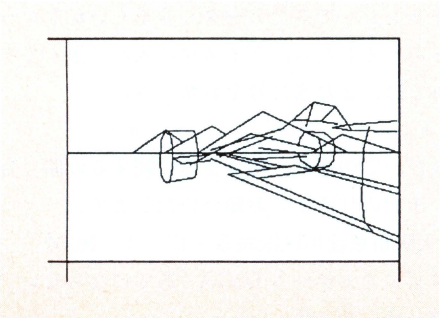


図10　10フレーム目の完成予想図

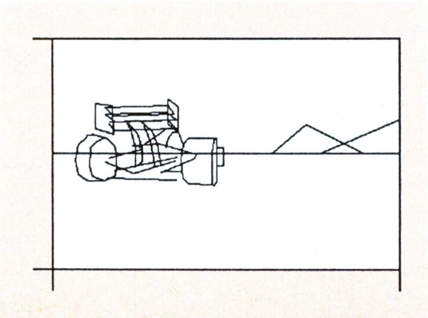
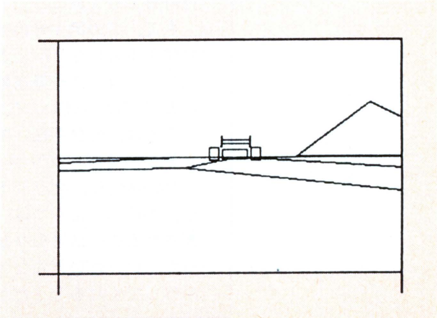


図11　40フレーム目の完成予想図



する

解説：アニメーションに，異常があるようでしたら，FFEを起動して，ロードして修正を加えるか，もう一度作りなおしてください。「CGA大学／修士課程」まで進んでいるような方は，フレームソースファイルを直接直すほうが早いでしょう。作画するとき，1から5フレーム目までは，アップにしすぎて少し変な画像になりますが，それで問題ありません。

連続アニメーション

今回制作した3カットを連続してアニメーションさせてみましょう。各カットだけでは表現できない迫力が出ます。そのために，まずタイムチャートファイルを作ります。

コマンドラインから実行する場合は，

mktch x10a001 x10b001 x10c001

とします。PESから実行するときは，コマンドウィンドウのMKTCHを選択し，画像ファイルウィンドウの「X10A001」「X10B001」「X10C001」を順番に指定して実行します。

その結果できるファイルが「X10A.TCH」です。このファイルの中身は，

```
.timechart
  x10a [1-20]
  x10b [1-5]
  x10c [1-40]
.endchart
```

となります。ここで，エディタを使える方は，

```
.timechart
.wait 20        x10a001
.wait 1         x10a [2-19]
.wait 10        x10a020
.wait 1         x10b [1-5]
                x10c [1-40]
.endchart
```

としてください。これは，カット「X10A」の最初と最後に“止め”を入れているのです。タイムチャートの詳しい文法は，マニュアルのT-251，Q-14をご覧ください。

タイムチャートファイルができたら，

HANIM X10A

で，連続アニメーションを表示します。

どうですか？　ちょっとしたものでしょう。いかにも，軽快に発進したという感じですね。この先，この車はどこに向かうのか？　このアニメーションに続くカットは，皆さんの手で作ってください。

おわりに

さて，いかがでしたでしょうか？　FFEは決して操作性がよいとはいえませんが，それでもたったこれだけの操作で結構かっこいいアニメーションが制作できるのです。文月さんが，“TORNADOは誰にでもできるんだ”といったのもまんざらウソじゃない，確かにこれなら自分でもできそうだと思いましたか？　でも，まあちょっと待ってください。その結論を出すのは，まだ少し早すぎますから。

ということで，ぜひ上記に続くカットを制作して，作品にまとめてみてください。残念ながら，大部分の方がうまくいかないのではないかと予想します。そのとき，“オレには，やっぱり才能がないんだー”などと嘆く必要はありません。うまくいかない原因は，だいたい誰でも同じで，その解決法もおおよそ確立(?)されています。

そこで，次回では，うまくいかない原因と，その解決法について解説します。前後編だった予定がだんだん長くなりますね。

さて，この連載では，ご覧のように，CGAを具体的に制作しながら解説していきますが，皆さんは，どのようなCGAを制作してみたいですか？　“紅の豚のような空中戦闘シーン”“ずばりSTAR WARS”“やっぱりバトルロボット（これだけはやめてくれー）”“フライングロゴやアイキャッチ”……。この連載で取り上げてほしい内容がありましたら，ぜひお便りください。また，その際，ご自分でチャレンジしてみたデータ，うまくいかなかったデータなどありましたら，ディスクに入れて送ってください。それでは，また来月号をお楽しみに。

*　　*　　*

当チーム及び，この連載，各コラムに関するお問い合わせなどは，下記の住所までお願いいたします。

〒533　大阪市東淀川区淡路5－17－2　102号
プロジェクトチームDōGA「あてにならないアフターサービス係」

CGAマガジン編集部より

年内発表に向けて活動を開始した“CGAシステムデータ集「DōGA CGAマガジン」”ですが，いきなり年内発表がむずかしい状態になっています。その理由は……，“ぜんぜん投稿データが足りないぞ～！”。このディスクマガジンは，ユーザーが作ったデータの発表の場として設けたんだから，投稿データがないと作りようがない。ということで，みんなガンバるんだ。いちおう，第1号の投稿締め切りは，10月31日としておこう。

さて，この第1号の特集（DōGA CGAマガジン編集部内で制作する分）を「F1」とすることにしました。車本体だけでなく，コースや障害物，ピットなどに力を入れたいと思います。CADは使いこなせるが，作りたいネタが特にないという方は，ぜひ手伝ってください。連絡お待ちしています。

そういえば，グリフォンや零式で有名なダークサイド「TOSAKA」に，オリジナルロボットを投稿してって頼んだら，先日作ったやつは，1体でディスク数枚になるし，作画させるのにはメモリが8Mバイトいるって断られてしまった。いったいどんなデータだ。でも，兵器モノをいくつか作ってくれるそうだから，お楽しみに。

メモリ食い食いウラ画面

Ogikubo Kei 荻窪 圭

やっと6Mバイトにしたけど，ちゃんと目的があってのこと。「MATIER」のサンプル版を存分に試してみたい，というのがその理由。扱うデータの大きさを考えると，メモリ食うのもしかたないでしょう。

残暑ざんしょ。などといっているうちに，いまは秋なのだが，残暑。

さて，最近冷遇されていた我が家のX68000。MacintoshIIcx, PowerBook100, DELL 486Dと軒並み8MバイトのRAMを突っ込んでいたのに対し，X68000だけはわずか2Mバイト。むごい仕打ちであった。雨の日も風の日も働き，10日間電源入れっ放しのときもあり，今年は熱暴走もしなかったのに，なのにメモリは2Mバイト。

しかしである。とうとう買ってしまったのだ。増設メモリというやつを。いまやSIMMモジュールは1Mバイトあたり5,000円といわれる相場のなか，非主流派マシンの悲劇か，コストパフォーマンスの低いメモリではあったが，我が家の初代X68000も合計6Mバイトになったのだ。

X68000のメモリ増設もSIMMだったらなあと。だったら台湾かどこかで作られたIBM PC用とかMacintosh用の安いメモリを使えたのに。いまはとにかく，東南アジアパワーで枯れた技術はどんどん安くなっているから，なめてはいけないのだ。

で，「どうして今頃，荻窪圭はメモリを増設なんかしたのだ」という声があると思う。今頃，だ。やはり，メモリを増設するからにはそれなりの理由があるわけである。「シムアース」をやりたいけど2Mバイトじゃあ意味がない，とか，でかいRAMディスクでもないとコンパイルが遅くてしようがない，とか，最低6MバイトはないとSX-WINDOWなんてかったるくてやってらんねえぜ，とか，ショップのおじさんに説得されて気の弱い少年は4MバイトのRAMを買ってしまった，とか，霊感商法にやられた，とか理由はいろいろだ。

私の場合はつまり，「MATIER」である。マチエール。英語でいうとマテリアル。こいつが面白いのだ。しかし，2Mバイトだと裏画面さえ使えないし，スクリーンセイバーさえ動かないし，いちいちリセットするのも面倒で，ついつい足が遠のいてしまう。いかん，娯楽のためなら三千里，こんなに遊べるソフトがあるのに，メモリが足りないなどという軟弱な理由で使えないとは悲しいではないか。

で，奮発。感覚的には8Mバイトくらい当たり前となっているから，どうせ買うなら4Mバイトボードを。というわけで，合計6Mバイトになったわけだ。そうなってくると，40Mバイトのハードディスクってのがさびしさをほとばしらせてくれるのだが，さらにSCSIボードを買ってハードディスクを買って，となると話がややこしくなるため，もうしばらく我慢，である。

あんまり我慢するのもよくないのだけれどね。貧すればDOS，じゃなくて，貧すれば鈍すっていうくらいだから。

6Mバイトに増えた日

秋葉原へ行き，アイ・オー・データ機器の4Mバイト増設RAMボードを買ってくる。X68000の周辺を片づける。これはおおごとである。書類やゴミの山からX68000が剝き出しになる。裏に回るスペースがとれないので（昔はキャスター付きのラックで，ゴロゴロと手前に移動し，裏に回って作業すれば楽ちん，だったのだが），X68000を手前に引き出す。

引き出せない。思えば，X68000の裏からはデッドテックなほど，ゲージュツ的なほど，さまざまなケーブルが生えており，なかには長さがギリギリのもの，うっとうしいからといって束ねてしまったものもあり，うかつに力を入れるわけにもいかない。

そろそろと10cmほど手前に引き出す。周りの本を除け，やっと腕が入るスペースを作る。右腕を本体の裏のほうへ差し込み，ケーブルをひとつずつ抜く。電源ケーブルが本体直付けなのが悩ましい。こいつだけは，外せないのだ。

おっと，忘れてはいけない。背面のメイン電源をOFFにしなければ。

おおかたのケーブルを外して，X68000をもう少し引き出し，横に向ける。なんとか，ボードを差し込むだけのスペースを確保。ここで，まったく使われていないスロットの片方を開ける。

箱から増設RAMボードを取り出す。その前にいちおう縁起モノなので，金属剝き出しのものに触り，身体に溜まった静電気を放出する。箱からボードを取り出す。メモリがぎっしりと（というほどでもないが）並んでいる。

方向を間違えないようにボードを差し込む。はまらない。強く押し込む。まだ駄目だ。さらに強く渾身の力と愛を込める。ザクっとはまる。さらに押す。バキっと壊れる。んなことはない。

再び，腕を伸ばして，モニタケーブル，プリンタケーブル，HDDケーブル，TV CONTROLケーブル，IMAGE入力ケーブル，半透明用ケーブル，RS-232Cケーブル……もうないよな……を再接続。あー，面倒臭い。なんでこんなにケーブルがいるんだか。

でも，SCSIしてたり外付けFDDつけたりMIDIしたりすれば，こんなもんですまないわけで，あわわわわ。こんなときに思うのだ。パソコンの各種コネクタが，全部前面についてたら壮観だろうな，って。フロッピーディスクドライブのスロット以外は全部コネクタで埋まっているパソコン。

すべての接続が終わって，裏のメイン電源を入れる。でもって，ハードディスクをONにし，本体をONにする。起動する。いつもと何も変わらない。当たり前だ。

腕を組んでちょっと悩む。ここんとこずっと同じ環境でX68000を触っていたものだから，搭載メモリの変更をするコマンドをちょいと忘れちまったのだ。

SWITCHなんだよな。これを忘れてはいけない。で，SWITCHコマンドでメモリを“6144KB”に設定する。再起動する。

ここからが楽しい時間だ。

MEMFREEする。ふふふ。フリーエリアが5Mバイト以上ある。しかし，これは最適なメモリ利用をするために不要なデバイスドライバを外しての値だ。メモリが6Mバイトもあれば，そんな気遣いは無用。

外していたRAMディスクドライバを復活させる。しかし，身についた貧乏症はなかなか取れぬもの。テンポラリとして256Kバイト確保するにとどめる。

続いて，「Communication SX-68K」に必要なRS-232Cドライバ，RSDRV.SYSを入れ，ついでに，プリンタドライバも入れる。FIXER4のバッファの設定を増やす，CONFIG.SYSのBUFFERSを増やす。あとは，PCMDRV.SYS，OPMDRV.SYS，FLOAT2.X，FIXER.SYS，HISTORY.X，IOCS.X，HCDRV.SYSだ。最後のやつは，X68000のドライブでIBM PCフォーマット（1セクタ512バイトのフォーマット）2HDディスクを読むためのもの。

もっと便利なデバイスドライバがいろいろとあった気がするが，とりあえず，このあたりが妥当だろう。

6Mバイトの使い方

そもそも私は，リセットとかリスタートとか再起動とかって嫌いなのである。必要な環境は全部メモリにロードしておき，あとはOSっていうか，シェルを基盤に，行ったり来たりしたい。6Mバイトあればそれができる。必要なデバイスドライバや常駐ソフトを置き，チャイルドプロセスを駆使して親亀の背中に子亀を乗せて，さらに，SX-WINDOWさえ快適に動く。

さらなる目標は，ノンストップコンピューティングなのだが，寝ていたり出かけていたりする時間に動かしつづけるのも電気の無駄遣いなわけで，本当はX68000がレジューム機能をもってくれるのがいちばんいい。最近，ゲームもあまりやんなくなったしね。

6Mバイトもあれば，普通の人はRAMディスクをでっかくとったりするのだろうけど，RAMディスクをテンポラリファイル用として使うならともかく，たいていの場合は安全を期する意味もあって，ハードディスクとRAMディスクに同じファイルが存在することになる（っていうか，ハードディスクのファイルをRAMディスクにコピーし，それが更新されたらまたハードディスクに戻す，みたいな）。

なんか，無駄な気がするのだ，私は。RAMディスクでないと遅くてやってられない，ってなこともないし。

しかし，うれしいよね。MicroEMACSででっかいファイルを編集することもできるし（なにせ，1Mバイトのテキストファイルとかがあったもので），チャイルドプロセスであのメモリ食いの「MATIER」なんかが動いたりする。

でも，こうして遊んでいると，タスクスイッチング機能がほしくなってくるのもまた事実。マルチタスクなんていいから，オンメモリタスクスイッチング機能なんかつかないかなあ。ちまたで噂のDR-DOS/Vってのがあるのだが（デジタルリサーチ製のMS-DOS互換OSね），こいつのタスクスイッチング機能がMS-DOS 5.0よりずっと強力で，かなり便利なのだ。いつでもキー一発でタスクスイッチングメニューが出て，アクティブなタスクを変更できる。すると，いろいろと起動時間を気にせずに遊べて楽しいのだ。

で，話は最初に戻る。「MATIER」である。こいつはメモリがたくさんあれば，表画面＋裏画面×4＋アンドゥ画面を持つことができる。さらに原則として，起動時，終了時，チャイルドプロセス時にはグラフィック画面をクリアしないで表示したままでいる，という便利な特性を持っている。だから，チャイルドプロセスで「MATIER」がサポートしていない（製品版ではサポートするかもしれないが，よく知らない）GIFやJPEG，そして数多のフォーマットのデータを読み書きできるし，フィルタをかけて加工することもできるのだ。

湯水のようにメモリを使え。それで話がすむのなら，増設してやろう，という気にもなるではないか。

データの使い回し

「MATIER」みたいな面白いソフトが出てくると，いろいろと遊びたくなってくるのだが，私は川原由唯氏のような絵心も，心どころか技術もない。あるのは面白がる精神と，何台ものパソコンだけだ。

たとえば，Macintoshがあるが，Macintoshとて完璧ではない。68030の16MHzで，なおかつコプロまで搭載していながら，体感速度（スループットっていっていいのかな）はそんなでもないし，「MATIER」並みの機能をもつソフトになると，かなり高価（100,000円以上）になってしまうし，安いソフトでは「MATIER」ほどの遊び心がない。

最近思うのだが，68030/16MHzのMacintoshがあって，その隣に80486SX/20MHzのPC/AT互換機があって，その隣に68000/10MHzのX68000があるのだが，使い分けていて，それほどの違和感はない。そりゃあ，純粋にCPUパワーに頼る計算ものでは露骨に差が出るけれども，PC/ATをWindowsで使い，MacintoshをMacintosh OSで使い，X68000を普通にHuman68kで使う（ここがミソ）。すると，ものによっては，X68000がいちばん快適だったりする。ウィンドウシステムの重さ，ハードを直接叩くX68000のHuman68kベースのソフトの軽さ（と，そのソフトを作った人の技）はCPUパワーの違いを乗り越えてしまう。当たり前のことである。

で，話は戻って，グラフィックデータのやりとりである。

そもそも，昔，X68000が出た頃，まあ，いまでもそういう雰囲気が根強いのだが，画像データ＝そのコンピュータのディスプレイ能力に依存するもの，であった。決まって，大きさはグラフィック画面に依存し（PC-9801なら640×400，X68000なら768×512か512×512），色数も同様。CRTに表示する画像は，出力装置としてCRT以外のものを考えるなど及びもつかず，ハードウェアに密着した画像といってもよかった。

しかし，グラフィックデータがただ画面を賑わすだけのものではなく，ひとつの表現となり，印刷を前提としたイラストとなり，テキストデータと同様に流通するもの

になると，事情は変わってくる。

グラフィックデータもテキストデータと同様，ハードウェアから独立して流通するものなのである。

X68000でネックになるのはここだ。ドットの縦横比。MacintoshやPC/ATは640×480ドットで，これは画面上の1ドットの縦横比が1：1になるのだけれど，X68000は512×512ドットで横長のピクセルとなり，いろいろと不自由なのである。不自由。

さらに，ほかのマシンとX68000とでデータをやりとりしようと思うと，どうしても縦横比変換が必要になり，画質が落ちてしまう。残念である。いくらハードウェアから独立して流通するものだとはいえ，最低限のベースは必要で，それが縦横比1：1だといっていい。

続いて，色数の問題が出てくる。

いうまでもないことだが，モノクロ，16色，256色，32768色（65536色），1670万色と1ピクセルあたりの色数によって，用途や適した画像形式が変わってくる。Macintoshの世界では16色から順に4ビットカラー，8ビットカラー，16ビットカラー，24ビットカラーと呼んでいる。1ピクセルあたりに必要なデータ量で表すわけだ。32768色（X68000のように輝度ビットを使うことは希だ）なら15ビットカラーではないか，などと細かいことはいわないように。

1670万色をフルカラーと呼ぶのはMacintosh，トゥルーカラーと呼ぶのはPC/AT互換機界。PC/AT互換機界では，32758色はハイカラーと呼ぶようだ。

名前なんてどうでもいいか。

まあとにかく，ドットの縦横比が同じであれば，どのマシンで見ても相似形は保たれる。色数が同じであれば，どのマシンで見ても情報量は一定に保たれる。そういうことである。

ただ見るだけなら，多少の違いはものともしないいいかげんさを人間の目は持っているが，いざそのデータをいじってなんかしようと思うと，とたんにややこしくなってしまうのだ。

私はMacintoshで取り込んだりいじったりしたデータを，「MATIER」へもっていって遊びたいのである。いまのままだとちょっと面倒なのが残念なのだ。

ちなみに，両者のデータコンバートは，PC/AT互換機を中継ステーションにして行っている。MacintoshとX68000を直接繋いでもいいのだが，X68000のRS-232Cはモデム用で埋まっているため，MacintoshからPC/AT互換機の5インチディスクへシリアルケーブルを使って転送し，その5インチディスクをX68000へもってきて読み込む，ってなことをやっているのだ。X68000→Macintoshのときは逆の手順。ちょっと面倒だけど，まあ，しかたがない。

話は飛んでちょっと立体視

で，「MATIER」を使ってなにかしようか，と思っていたのだ。お絵描きでは川原氏の足元にも及ばないので，ここは自然画で遊んでみよう，と。「MATIER」のもつさまざまな機能は，自然画で遊ぶには申し分ないからね。わざわざメモリを6Mバイトにしたかいがあるってもんだ。

そんなとき読んでいた本が「脳天記」っていう，久住昌之と加藤総夫の共著。加藤氏が脳の専門家で，まあ，2人で脳をネタにヨタ話をする，っていう本なのだ。

いちばん面白かったのは記憶の話。カナダの外科医のセンセイが行った実験（いや，治療）の報告で，患者に局所麻酔をかけ，頭蓋骨を開けて，脳の表面を電気刺激する。すると，刺激する部位によって異なった記憶が出てくるのだ。ある場所だと幼少期の記憶，別の場所だと昔聞いた歌，という具合に。これは面白い。もし，頭蓋骨を開けたりせずに，脳の任意の場所を刺激できるようになったら，絶対に，バーチャルリアリティなんかより面白い体験ができる気がする。

で，これは本誌にはどうでもいい話で，「MATIER」とは関係ない。本題は，この「脳天記」に出てくる裸眼立体視のほうだ。どうして脳と裸眼立体視が関係あるのかというと，2枚の写真をひとつにしてあたかも3Dであるかのように見せているのは，脳である，ということなのだ。これを読んでいて，パソコンで裸眼立体視してみよう，と，ふと思ったしだいである。

そもそも，立体視なんてめずらしいことでもなんでもなくて，さまざまな業務で実際に使われているし（国土地理院の地図作成にも使われているそうだ。航空写真を2枚用意して，立体視するらしい），X68000にだって立体視端子があったくらいだ。ユーザーレベルでも立体視CGの愛好家は多く，NIFTY-Serveでも立体視用の画像データがたくさんアップされている。そういう先達たちがたくさんいるなかで，普通のことをやってもしようがない。

えっと，その前に，いちおう概略だけでも描いておこうか。

立体視ってのは，左目と右目に異なる画像を見せて，脳の中でそれをひとつに合体させるという作業をすることにより，左目に入った画像と右目に入った画像の差異があたかも立体的であるかのように見えてしまう技を使った快楽，である。

で，左目と右目に違う画像を入力するための方法がいろいろあって，赤青メガネってのがいちばん貧乏臭い。あのセロハンを張ったメガネはアバウトだし，だいたい青のセロハンの向こうに青い線がうっすらと見えたりして，気になるとどうしようもなくなって，立体どころではなくなる。

偏向レンズってのもある。マイケル・ジャクソンの「キャプテンE.O.」はこの方法だったと思う。しかし，実のところ，私のメガネは結構フレームがでかくて，そのうえにさらにメガネをかけるとなると，それが気になってしかたがない。

X68000の立体視端子などは液晶シャッターである。ゴーグルの左目と右目に液晶シャッターがついていて，右目用の絵と左目用の絵が画面に交互に（瞬時に）表示され，液晶シャッターは本体と連動して，右目，左目，それぞれに該当する画像しか見えないようにしている。

もうちょいおおがかりだと，ディスプレイ側に液晶シャッターをつける。で，それを偏向レンズメガネで見る。そうすると，液晶シャッター式と違い，大勢で見られるようになるわけだ。

いずれにしろ，大昔の映画のフィルムを見ているようなチカチカ感が味わえてよろしい。

以上は，強制的に左目と右目に別の画像を送り込む方法であった。が，裸眼立体視は違う。みずからの意志で左目と右目で違う画像を見るのである。これがなかなかむずかしくてよい。

ステレオ写真ってのは皆これで，簡単に

立体的に見られるようなビュワーもあるけれども，やはり，自分の意志で脳の無意識の動きを抑えて立体視してしまう，というほうが快感だ。

最近，流行っている（のかどうか知らないが）立体視は，裸眼立体視。なぜなら，誰でも簡単に作れるから。いちばん簡単なのは，同じ絵をカメラの位置をちょっと左右にずらして（数センチほど）2枚撮ればいい。で，焼き上がったら，2つを横に並べて，左目と右目でそれぞれを見る。

目というのは，少し離れて2つついている。たくさん離れている人とか少ししか離れていない人とかもいるけれども，大同小異だ。ということは，左目で見ているものと，右目で見ているものは微妙に角度が異なるわけで，それをシミュレーションしようというのが立体視だ。あくまでも，「シミュレーション」ってのがポイントで，もっと簡単にいえば，「脳をだまくらかしている」ってのが真相だ。

この，「立体的＝リアル」という枠組みを鼻で笑い，「脳をだます＝立体でないものまで立体的に見てしまおう」という面白がりのコンセプトのみが「脳天記」では貫かれている。もう，たいしたものである。だいたい，この面白がりの感覚，写真を面白がって遊ぶ感覚，南伸坊の「笑う写真」もそうだし，この「脳天記」もそうだし，このへんの結びつきを見ると，路上観察学会とか，フォトグラファー滝本淳助とか，白夜書房とかそういったキーワードが出てくる。まともではない。

要は左目用の画像と右目用の画像をずらしてやればいいじゃん，ってわけで，個人的にいちばん最高だったのは，川に浮かぶゴミ。いくら下流の川でも多少はたゆたっているわけで，同じ位置でちょっと時間をずらして撮影すると，ぷかぷか浮かんでいるゴミの位置が変わる。それを立体視すると，なんと，川の上にゴミが高くなり低くなりして浮いているように見えるのだ。これはシュールである。川のゴミってところが，なんとなく自然保護なんかと絡めた意味がありそうで，実はそんな意味なんてまったくない，ってところがいい。

CRTでの立体視

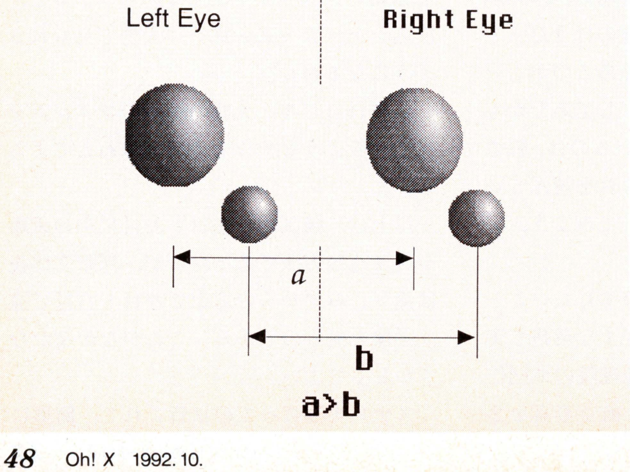
そんなわけであるが，パソコンのCRT上でやる立体視，となると，けっこう真面目なものが多くて，3Dソフトを使って視点をずらして作成した2枚の絵を立体視するとか，ステレオ写真をスキャンして並べて立体視とか，そんなものが中心のようだ。それじゃあつまらない，ってわけで，手作り立体視画像が来月のテーマだ。

いや，いきなり来月，っていわれても困るだろうけど，困ってください。今月はまだ準備ができていないのだ。なにせ，4MバイトのRAMボードを買ってきたのが2日前だもので。はい。今月は，それが本当にできるのか，1枚の取り込んだ自然画像を立体化できるのか，っていう実験で手一杯だったのだ。

図1がとりあえず「MATIER」で作ってみた立体球である。作成時間2分。いや，もっと短いかな。

だって，ねえ，大小2つの球を画面の左のほうに描き，でもって，それを右にコピーして，小さいほうの球をちょっと左へずらす。これだけだから，思ったより簡単。

図1



えっと，その前に断っておくと，裸眼視には平行法と交差法がある。平行法，ってのは，左の絵を左目で，右の絵を右目で見る，って方法で，交差法ってのは，左の絵を右目で，右の絵を左目で見るという方法。どっちが簡単か，っていうと，人によってそれぞれらしいのだが，平行法のほうがポピュラーだ。ステレオ写真機ってのは平行法だし，平行法だと，いざとなれば左の絵と右の絵の間に衝立を置けば，つい反対側の絵が気になってしまって立体視できない，っていう人への助け舟になる。

でも，いろいろと意見があって，交差法しかできない人とか，交差法だと大きな画像でも対応できるが，平行法だと小さな画像しか対応できない，って人もいる。

私は，最初に「平行法のほうが簡単だよな」って思ってしまったせいで，いまだに交差法ができない。まあ，CRTが相手だと，雑誌などの写真よりずっと大きな画像なわけで，相手が大きいと立体視がむずかしくなるのは事実のようだ。

そもそも，脳というのは，目の前にあるのが平面だと思うと，ちゃんとその平面に合わせてピント合わせと眼球の動きを制御しているわけで，それに逆らって脳をだまそうというのだから，立体視しやすい画像，しやすい大きさというのはちゃんとある。

で，図の解説。左目用の球2つと右目用の球2つを作る。両者はもちろん同じものだ。で，どちらを手前にあるように見せるかが問題だ。脳ってのはバカなのか利口なのか，まったく同じ画像でも，それを立体視の要領で左右の目で見て，それぞれの画像をうまく一致させると，それなりに立体に見えてしまうのだ。

まず，大きな球どうしの距離，小さな球どうしの距離に注目する。両者の距離が近いほうが手前に見える。それだけである。

＊　＊　＊

いやあ，今月はなんだったのか，半分来月の予告だったりして，これで来月，全然違うことを始めたら，大笑いだな。

えっと，来月の予告をすると，たとえば，ヌード写真がある，と。1枚しかない。これを立体視したい。胸が飛び出てくるとか，ね。そんなとき，同じ画像を左と右に作り，それをメッシュ変形なんかで，立体的に簡単に加工できないか，ということである。それは，できるのである。簡単に。ただ，雑誌にそういう画像を載せてもいいよ，っていう寛大な女の子がいないのだ。うーん，残念。誰か，いませんか？

それから，「MATIER」でバカをやる，っていうのも考えていたりする。そういうわけである。では，さようなら。

Oh!X LIVE Special

このコーナーが始まって丸5年間が過ぎ，6年目に突入しました。X68000ユーザーのMIDI所有率も25%近くになり，ますます音楽とは切っても切れないパソコン雑誌に成長したんじゃないかと，内心ほくそ笑んでいます。担当者も初心に戻り，頑張っていく所存です。これからも末ながく応援と投稿をよろしくお願いします。　(S.K.)

X1・MIDIシーケンサ用（CM-64対応）

美少女戦士セーラームーン

Hasunuma Masaru
蓮沼　勝

美少女戦士セーラームーン

月にかわっておしおきよ!

さて，X1ユーザーにはとびっきりのおしおきをしてあげましょう（なんじゃそりゃ）。小見出しを見て，ピーンときた人もいることでしょう。「美少女戦士セーラームーン」の挿入歌を5曲まとめてお送りします。

CM-64用の曲ということで，またもやMIDI BASICが必要になります。また，変身のテーマだけはギターカード（7番）が必要ですので注意してください。

知らない人のために情報をお届けしましょう。関東地方ではテレビ朝日で土曜日の夜7時から放映されているアニメなのです。なんでも「あの筋」では爆発的な人気だとか。主人公の女の子は「月野うさぎ」ちゃんという怪しげな名前で，「ムーン・プリズム・パワー・メイクアップ」とか唱えながら，セーラームーンに変身するのです。ちなみにお友達の“天才少女”水野亜美ちゃんはセーラーマーキュリーに，“霊感少女”火野レイちゃんはセーラーマーズに変身します。原作ではセーラージュピターとセーラービーナスもいるそうな。

さて，曲は5曲もありますが，リストをみれば一目瞭然といった感じで非常に短いものが5曲です。これは打ち込みやすいのではないでしょうか。それぞれの曲にはわかりやすいタイトルがついていませんでしたので，目安となるように曲目を書いておきましょう。

1)　変身のテーマ（要ギターカード）
2)　名乗りをあげるテーマ
3)　ムーンティアラのテーマ
4)　アイキャッチ
5)　変装のテーマ

てな具合です。特に「変身のテーマ」はコーラスがないことを除けば，原曲よりかっこよく聴こえます。若干のアレンジが入っているような感じですね（X68000で聴きたいという人は59ページを参照）。

ちなみにこの投稿にはほとんどコメントがついていなかったので，非常に苦労しました。ファイルネームだけが曲の手がかりなのですから。これがいまをときめく「セーラームーン」なんかじゃなくって，「魔女っ子チックル」の挿入歌でも送られていた日には泣きを見るのはまず確実です（要するにボツ）。蓮沼君，曲を送るときはコメントをちゃんと書いてくださいね。

リスト1

日本音楽著作権協会(出)許諾第9271376-201号

```
10 'SAVE    "S-Mo 10-1    .C7 "            '92/07/09 THU 14:19
20 CLS0:TEMPO0
30 A$="I101^7,100P8¥128R":FORI=2TO16:PLAY"M"+MID$(STR$(I),2)+A$+
":";:NEXT:PLAY""
40 '        P=LA/PCM (0/1)  M=MODE(0-3)  T=TIME(0-7)  L=LEVEL(0-7
)
50 PLAY "T148M16I149V110^64,64O3L4B>EA>D:M12I149V120:M13I135V70:
M10I165O2L4V110:M10I165O2L4
60 PLAY "R1&R2.^64,0:L4O5G1&Q7G2.&G8.I162:L4O6RC2.&Q7C2.&C8.I137
 :CCCCCCCD:M10I101O2L8R1R>CCC<BBG4:M10I101:R1R2.M11I116:M14I212
70 FORI=1TO2
80 PLAY ":P4V100O5L4Q4ER2.R2AG:P10V80L4O5Q4BR2.R2AG:L4CDC8C8DCDC
C:O2L8F+F+F+F+F+F+F+F+F+F+F+>F+RF+R<:V120L4O3C+RRRRRC+C+:O2L8E
RRDE<BR>DR<B>DEARGR:O3L8ERRDE<BR>DR<B>DEA4G4:M3I106O5V70L8ERRDE<
BR>DR<B>DEARGR
90 PLAY ":ER2.R2<B>D:BR2.R2B>D<:CDC8C8D8CC8DCC:F+F+F+F+F+F+F+F+F
+F+F+F+>F+RF+R<:C+RRR8C+R4.C+C+:ERRDE<BRA&A2B4>D4:ERRDE<BRA&^1,2
55A2^1,0B4>D4:ERRDE<BRA&A2B4>D4
100 NEXT
110 PLAY ":Q8L8EDGE4.RDEDGE4.RD:Q8O6E1&E1&:CDCDCDCD:G+F+>F+<F+G+
F+>F+<F+G+F+>F+<F+G+F+>F+<F+:C+:Q4L8EEEEEEEEDDDDDDDD:L8O2V70Q5EE
EEEEEEDDDDDDDD:M15I147V50O3E1D1
120 PLAY ":EDEG4AQ7G4Q8F+RRG+&G+2:E1&E2R8<V90I135L24E16F16F+GG+A
A+B>:CDCDCD8C>C8C16C16<G8:G+F+>F+<F+G+F+>F+<F+>F+<F+>F+F+<F+F+G+
F+:R1C+4.C+:CCCCCCCQ3E4.Q8E&E2:CCCCCCCERRE&E2:C1ERRE&E2
130 PLAY "I134O3V40C1<B1>::O6Q8L8C2.<EFF+2&F+8&¥139F+32&¥150F+32
&¥161F+32&¥172F+32>¥128F+4:CDCDCDCD:F+F+>F+<F+F+F+>F+<F+F+F+>F+<
F+F+F+>F+<F+:C+:Q4CCCCCCCC<BBBBBBBB>:L16V55O3Q8CRC8CRC8CRC8CRC8<
BRB8BRB8BRB8BRB8>";
140 PLAY ":M3I145O5V50G1F+1:M4I123P13V110O5L8C2.<EFF+2.>F+:M5I14
5O5V50E1E-1
150 PLAY "<A1B1>::D1E2R<L24E16F16F+GG+AA+BL8>:CDCDCDC8D8D:F+F+>F
+<F+F+F+>F+<F+F+F+>F+<F+F+F+>F+<F+:R1R2R8C+8C+:<AAAAAAAABBBBBBBB
>:<ARA8ARA8ARA8ARA8BRB8BRB8BRB8BRB8>:A4G8B8&B2&B1:D1E1:F+4E8F+8&
F+2A-1
160 PLAY "C1<B1>::Q8L8C2.<EFF+2&F+8&¥139F+32&¥150F+32&¥161F+32&¥
172F+32>¥128F+4:CDCDCDCD:F+F+>F+<F+F+F+>F+<F+F+F+>F+<F+F+F+>F+<F
+:C+:Q4CCCCCCCC<BBBBBBBB>:CRC8CRC8CRC8CRC8<BRB8BRB8BRB8BRB8>::C2
.<EFF+2.>F+
170 PLAY "<A1>::D2&D<BF+D>:CDC8>C8<G16F8.:F+F+>F+<F+F+F+>F+16F+1
6<:O2R2.D16D16:<AAAAAAAA>:<ARA8ARA8ARA8ARA8>::D1
180 PLAY ":P8GRRARRB4&B2.:P8V80I149O5E RRE RRE-4&E-2. :L8CRRCRRC
R:>F+<F+F+>F+<G+F+A+:O3C+2.C+:Q8ERRERRE-4&E-1:L8Q8ERRERRE-4&E1
190 FORJ=1TO2
200 FORI=1TO2
210 PLAY ":P4V100O5L4Q4ER2.R2AG:I137P10V80L4O5Q4BR2.R2AG:L4CDC8C
8DCDCC:O2L8F+F+F+F+F+F+F+F+F+F+F+>F+RF+R<:V120L4O3C+RRRRRC+C+:
Q8O2L8ERRDE<BR>DR<B>DEARGR:V100Q8O3L8ERRDE<BR>DR<B>DEA4G4:M3I106
O5V70L8ERRDE<BR>DR<B>DEARGR
220 PLAY ":ER2.R2<B>D:BR2.R2B>D<:CDC8C8D8CC8DCC:F+F+F+F+F+F+F+F+
F+F+F+F+>F+RF+R<:C+RRR8C+R4.C+C+:ERRDE<BRA&A2B4>D4:ERRDE<BRA&^1,
255A2^1,0B4>D4V65:ERRDE<BRA&A2B4>D4
230 NEXT
240 PLAY "I134O3V40C1<B1>::O6Q8L8C2.<EFF+2&F+8&¥139F+32&¥150F+32
&¥161F+32&¥172F+32>¥128F+4:CDCDCDCD:F+F+>F+<F+F+F+>F+<F+F+F+>F+<
F+F+F+>F+<F+:C+:Q4CCCCCCCC<BBBBBBBB>:L16V55O3Q8CRC8CRC8CRC8CRC8<
BRB8BRB8BRB8BRB8>";
250 PLAY ":M3I145O5V50G1F+1:M4I123P13V110O5L8C2.<EFF+2.>F+:M5I14
5O5V50E1E-1
260 PLAY "<A1B1>::D1E2R<L24E16F16F+GG+AA+BL8>:CDCDCDC8D8D:F+F+>F
+<F+F+F+>F+<F+F+F+>F+<F+F+F+>F+<F+:R1R2R8C+8C+:<AAAAAAAABBBBBBBB
>:<ARA8ARA8ARA8ARA8BRB8BRB8BRB8BRB8>:A4G8B8&B2&B1:D1E1:F+4E8F+8&
F+2A-1
270 PLAY "C1<B1>::Q8L8C2.<EFF+2&F+8&¥139F+32&¥150F+32&¥161F+32&¥
172F+32>¥128F+4:CDCDCDCD:F+F+>F+<F+F+F+>F+<F+F+F+>F+<F+F+F+>F+<F
+:C+:Q4CCCCCCCC<BBBBBBBB>:CRC8CRC8CRC8CRC8<BRB8BRB8BRB8BRB8>::C2
.<EFF+2.>F+
280 PLAY "<A1B1>:R1R4Q8F+2.:D1E-1:CDCDCDC8D8D8C8:F+F+>F+<F+F+F+>
F+<F+F+F+>F+<F+F+>F+F+<F+::<AAAAAAAABBBBBBBB>:<ARA8ARA8ARA8ARA8B
RB8BRB8BRB8BRB8>::D1E-1
290 NEXT
300 PLAY ":L8Q5DERERRDERERR:L8Q5DERERRDERERR:L8DDRDRRDDRDR:L8>F+
F+RF+RRF+F+RF+RR<:O2L8CCRCRRCCRCRR:DERERRDERERR:Q4O3L8DERERRDERE
RR
310 PLAY ":<Q8E1&E1>:Q8<<E1&E1>>:D:>C+<:C:Q8E1&E1:Q8<E1&E1>
```

リスト2

```
10 'SAVE    "S-Mo 5-5      .    "           '92/06/19 FRI 13:13
20 CLS0:TEMPO0
30 A$="I101^1,0^7,100P8¥128R":FORI=2TO16:PLAY"M"+MID$(STR$(I),2)
+A$+":";:NEXT:PLAY""
40 '         P=LA/PCM (0/1)  M=MODE(0-3)  T=TIME(0-7)  L=LEVEL(0-7
)
50 PLAY "T126M2I193RV100O5L32GAB>CDEFF+<:M11I137RO5L32GAB>CDEFF+
<:M12I137:M3I213:M10I101:M10I101:M10I101:M13I162:M14I150
60 FORI=1TO2
70 PLAY ">G16<:>G16<:O3L16G:V110O3C:O2L8D4DDD.D.D:O3L8F+4F+F+F+.
F+.F+:O2L4>C+<F+F+F+:O5L16Q7R4DRERF+8.F+RRE:O5L16Q7GRRRDRERF+8.F
+RRE
80 PLAY "R2.GAB>CDEFF+<:R2.GAB>CDEFF+<:R2CCRC8.DR:R2C16C8C8.C8:R
4DDD16DD.D:R4F+F+F+16F+F+.F+:F+F+F+F+:R4ERF+RGGRG8.A:R4ERF+RGGRG
8.F+
90 NEXT
100 PLAY ">G16R2.R16G16<:>G16R2.R16G16<:GR2.R16G:C:R4DDD.D.D:F+4
F+F+F+.F+.F+ :>C+<F+F+F+8.F+16:R4FRFRE8.F8.:R4FRFRE8.F8.
110 PLAY "T126>G16R2.T110R8.T90G16<:>G16R2.R8.G16<:GR2.R8.G:CRRR
C:V100R4DDD.D16&D4D:F+4F+F+F+.F+16&F+4F+:>C+<F+F+F+F+:R4FRFRE8.F
4RG:R4FRFRE8.F4RG
```

リスト3

```
10 'SAVE    "S-Mo 10-3     .    "           '92/06/19 FRI 14:03
20 CLS0:TEMPO0
30 A$="I101^1,0^7,100P8¥128R":FORI=2TO16:PLAY"M"+MID$(STR$(I),2)
+A$+":";:NEXT:PLAY""
40 '         P=LA/PCM (0/1)  M=MODE(0-3)  T=TIME(0-7)  L=LEVEL(0-7
)
50 PLAY "T137M11I136P9V90RO6<L24AA+B>L16CC+L8P10D2.ED:M12I122V80
O2RG16&¥117G16&¥106G16&¥95G16¥128O1L16G8>GR<G8>GR<G8>GR<G8>GR<:M
2I123O6RR4L8D2.ED:M13I147V80O4RR4<B1>:M10I165RO2L4RCDCD:M10I165R
O2L8R4>C+<F+G+F+F+F+G+F+:M14I134P6V70RO5R4L8D2.ED
60 PLAY "F4.E16E-16D2:A8>AR<A8>AR<A8>AR<A8>AR<:F2D2:RDA6G6F6:CDC
D:F+F+G+F+F+F+G+F+:F4.E16E-16D2
70 PLAY "C2.DCE-4.D16D-16C2:G8>GR<G8>GR<G8>GR<G8>GR<E8>ER<E8>ER<
E8>ER<E8>ER<:C2.DCE-2C2:E1RC4A6G6E-6:CDCDCDCD:F+F+G+F+F+F+G+F+F+
F+G+F+F+F+G+F+:C2.DCE-4.D16D-16C2
80 PLAY "<B2.&B16_10B-16_A16_A-16V90E4F+4G4G+4>:B8>BR<B8>BR<B8>B
R<B8>BR E8>ER<E8>ER<E8>ER<E8>ER<<:<B1E4F+4G4G+4>:D1RE-DD-:CDCDCD
CD:F+F+G+F+F+F+G+F+F+F+G+F+F+F+G+F+:<B2.&B16_10B-16_A16_A-16V70E
4F+4G4G+4>
90 PLAY "<A1>:A8>AR<A8>AR<A8>AR<A8>AR<:<A1>:C2>C2<:CDC8D8D16D16B
8:F+F+G+F+F+F+G+F+:<A1>
100 PLAY "Q6D4.D4.Q8D4&R1&R2&R8:>A4.G4.F+4&R1&R2&R8<:D4.D4.D4&R1
&R2&R8:R1RAB2&R2&R8:O2C4.C4.C8R4O6D2:G+4.G+4.G+4R1R2G+:V50Q6F4.G
4.Q8F+4&R1&R2&R8
```

リスト4

```
10 'SAVE    "S-Mo 10-9     .    "           '92/06/10 WED 09:02
20 CLS0:TEMPO0
30 A$="I101^7,100P8¥128R":FORI=2TO16:PLAY"M"+MID$(STR$(I),2)+A$+
":";:NEXT:PLAY""
40 '         P=LA/PCM (0/1)  M=MODE(0-3)  T=TIME(0-7)  L=LEVEL(0-7
)
50 PLAY "T140M2I189^64,64O3L8B>_4ET145_A>_DT150_G>_C2.&:M11I136:
M11I136
60 PLAY "T162R16&R4^64,0:O6L8¥73C16&¥84C16&¥95C16&¥106C16&¥117C1
6&¥128Q7C2.<EF>::M10I101O2R16D4L4CDCD:M10I101O2R16G+4L8>C+<F+G+F
+F+F+G+F+:M12I118O1R16R4L8AAAAAAAA:M3I174V30O4R16R4C1&:M4I174V30
O3R16R4F+1&:M13I111V80O4L8R16R4EEE.EEE16EE
70 PLAY ":<Q7F+2.>Q6C4::CDCD:F+F+G+F+F+F+G+F+:A-A-A-A-A-A-A-A-:C
1:F+1:E-E-E-.E-E-E-16E-E-
80 PLAY ":<Q6A4Q4B8Q8B8&B1>:O5Q6F+4Q4G8Q8G8&G1:DC8D:F+G+F+>C+<:Q
7F+4Q5G8Q8G8&G1:<Q6A4Q4B8Q8B8&B1>:<Q6F+4Q4G8Q8G8&G1>:F+4G8G8&G1
```

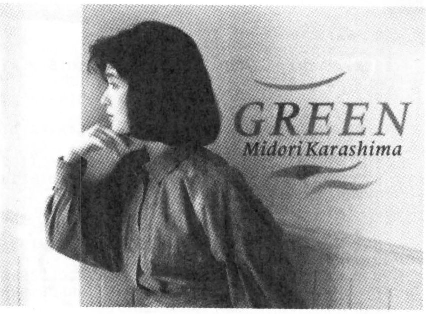
リスト5

```
10 'SAVE    "S-Mo 10-10    .    "           '92/06/19 FRI 12:00
20 CLS0:TEMPO0
30 A$="I101^1,0^7,100P8¥128R":FORI=2TO16:PLAY"M"+MID$(STR$(I),2)
+A$+":";:NEXT:PLAY""
40 '         P=LA/PCM (0/1)  M=MODE(0-3)  T=TIME(0-7)  L=LEVEL(0-7
)
50 PLAY "T130M11I137O5Q7E1Q8:M10I101V122O2L8D>C24<A24F24CD>C24<A
24F24RD4:M12I136O4Q6E1Q8:M13I134:M10I101V95:M10I101V110:M14I118R
2.O1E4:M2 I161:M16I162:M16I162
60 FORL=1TO2
70 FORI=1TO2
80 PLAY "P6O5L16<AB>CDER<AB>CDERERAR:L4CDCD:P10O5L16<AB>CDER<AB>
CDERERAR:P6V60O3A1:O2L8>C+<F+G+F+G+F+G+F+:O4L8EED+EEE16D16DE:Q8O
1L8ARRAA8R16A16Q4AAQ8:P5V60O4L16R8A_30AA~ARRA_AA~BRRBR
90 PLAY "Q6F+8.Q8D&D4&D4.R8:CDCD:Q6F+8.Q8D&D4&D4.Q8:V75F+1:G+F+G
+F+G+F+G+F+:EED+EEE16D16DE:Q6F+RREF+8R16F+16Q4F+F+Q8:RRA_AA~ARRA
_AA~ARRAR
100 PLAY "<AB>CDER<AB>CDERERAR:CDCD:<AB>CDER<AB>CDERERAR:E1:G+F+
G+F+G+F+G+F+:EED+EEE16D16DE:ARRAA8R16A16Q4AAQ8:R8A_AA~ARRA_AA~BR
RBR
110 PLAY "Q6F+8.Q8D&D4&D4.R8:CDC8.D16D8C8:Q6F+8.Q8D&D4&D4.Q8:F+1
:G+F+G+F+G+F+F+R :EED+EEE16D16DE:Q6F+RREF+8R16F+16Q4F+F+Q8:RRA_A
A~ARRA_AA~ARRAR
120 PLAY ">C<BAGAERD8.ERRRE:CDCD:>C<BAGAERD8.ERRRE:E1:>C+<F+G+F+
G+F+G+F+:EEDEDEDE:L16A8R8.A8.ARRARRA:R2ARRA
130 PLAY ">C<BAGAGRA8.>CR<RRE:CDCD:>C<BAGAGRA8.>CRRR<E:F+1:F+F+G
+F+G+F+G+F+:EEDEEEDE:RRERRERRE8RRRRE:A8.ARRA8R4.A
140 PLAY "F2A8.>C8.F8<:CDCD:F2A8.>C8.F8<:F1:>C+<F+G+F+G+F+G+F+:E
EDE16D16DEDE:L8Q4DDDDDDDD:RRARA8_AA~AR_AR~AR_AR~
150 PLAY ">E1<:CD8C8D16D8C16D8A8:>E1<:E8R4E8&E2:>C+<G+F+F+:EEDEE
16ED16EE:ERRQ8E&E2:BRR4B:O5L8Q4ERRQ8G+&G+4:O5L8Q4G+RRQ8B&B4
160 NEXT
170 END
180 NEXT
```

X68000・Z-MUSIC＋ZPP用（SC-55対応）

笑顔を探して

Okamoto Masakazu
岡本 正和



辛島美登里

辛島のおねえさま

さて，今月のイチオシの時間がやってまいりました。辛島美登里さんの曲で「笑顔を探して」です。最新よりひとつ前のアルバム「GREEN」からの選曲になります。Z-MUSIC＋SC-55用の曲ですが，さらに今月号の特集で紹介されている“ZPP.X”が必要になります。ぜひとも入力して活用してください。この曲を聴くためだけに入力する価値もあると思いますよ。ZPPの威力にも注目してください。

以前にも書いたように，私は辛島のおねえさまが結構気に入っているのです。だから，ちょっとやそっとの作品では載らないことになりそうですが，正直いって，この作品にはぶっとびました。SC-55から奏でられたイントロは「CDを聴いてるのかな？」と勘違いして，アンプのセレクタを押し直したほどです。どのくらいすごいかは想像できるでしょ。

ちなみにCDとの聴き比べもやりましたが，ほとんど遜色なかったです。しいていえば，ギターのソロに違和感がありますが，ほかでのカバーで十分すぎるほどのおつりがくるのでよしとしましょう。

ボーカル命ともいえる辛島美登里の曲を作るうえで，声をどんな音色で表現するかは最重要課題といえますが，この作品ではフルートを使用しています。彼女の声質か

らして，これは正しい選択だったといえるでしょう。私もきっと同じ系統の音色を選ぶと思います。

この作品はなかなか凝っていて，曲の途中でSC-55のエフェクトを変えるなど，MIDIを有効に使っています。参考にするとよいでしょう。

作者の岡本君は，今月号の“ZPP.X”や1992年4月号に掲載された「あじさいのうた」の作者でもあります。プログラムも作れて曲も作れる，う～ん，ストロング。これからもビシバシ投稿してくださいね。

リスト6

日本音楽著作権協会(出)許諾第927136-201号

```
==================  egao03.ZPP  ==================
.comment /       笑顔を探して(GREENより)
.comment /           作詞/作曲   辛島美登里
.comment /           プログラム   岡本正和
.comment /         midi-module  SC-55
.comment /           for Z-muSiC system
/==============================================================
/Z-muSiC init.
(i)(b1)(d1)              /  つい先日、単行本の21巻まとめ
                         / 買いという豪快なことをやってし
/ Track setup.           / まいました。(どこが豪快だとい
                         / う気もする。)いやあ、あんなに
/ Drums.                 / 店員の人にちやほやされたのは初
(m01,2000)(aMidi10,01)/ めてです。その漫画に関連したポ
(m02,2000)(aMidi10,02)/ スターやホストカードやら、しま
(m03,2000)(aMidi10,03)/ いには垂れ幕までつけてくれるあ
                         / りさまです。(さすがに垂れ幕は
/ Melody.                / もらわなかった。)またやってみ
(m10,2000)(aMidi02,10)/ ようかなと思ってしまいました。
                         / (こんなことだからお金がたまら
/ Piano.                 / ない。)
(m20,4000)(aMidi12,20)/  さて、曲の方ですが、まとめ買
(m21,3000)(aMidi12,21)/ いした本のエンディングテーマだ
                         / った曲です。 ZPPのサンプルとし
/ Strings,Brass.         / てはユーザーを選びますが、その
(m30,3000)(aMidi05,30)/ 分出来は上々なので許して下さい。
(m31,3000)(aMidi06,31)/ (実はFM音源に関してはトーシロー
                         / なので音色が作れないのです。)
/ Guiter.                / 今回のテーマとして、なるべく短
(m40,2000)(aMidi09,40)/ いリストで、どれだけ複雑なこと
                         / ができるかというものがありまし
/ Brightness,etc...      / た。結果として、なんとか 350行
(m50,2000)(aMidi07,50)/ を切ることができました。当然、
(m51,2000)(aMidi08,51)/ 相当音源の力に頼っていたりしま
                         / すが・・・。
/ Bass.                  /  また曲を送りますのでよろしく
(m60,2000)(aMidi01,60)/ お願いします!!
/--------------------------------------------------------------
/ SC-55 System SETUP /都合の悪い方は削除のこと、ただしそ
                     /の場合それぞれのトラックの初期化が
                     /面倒です。
(t1) @I$41,$10,$42                              / GS ID
(t1) X$40,$00,$7f,0                             / GS reset
(t1) X$40,$01,$30,3                             / reverb
(t1) X$40,$01,$38,1                             / chorus
(t1) X$40,$01,$10,   3, 0,1,0, 0,1,2,2, 2,0,0,6, 0,0,0,0
/ V_RESERVE    part. D  1 2 3  4 5 6 7  8 91112 13141516
/==============================================================
/ MML dataset
(t1)    t122
#ext_on /MML拡張展開を始めます。
(d0,0) /(第一、第二パラメータの[!] のみ有効にする。)
(#I,"kse-") / キースケールをE♭に設定します。
(#U,"@u+5")(#D,"@u-5")        / ベロシティアップ/ダウン
(#A,"q8")(#B,"q6")(#C,"q5") / ゲートタイム
/ Melody.
(t10) I[p0]y91,80y93,00@y1,$66,80@y1,9,65@74@h48@m30 r4
(t10) [r0][a][a5][a6][a][a7][a3][a4][a8]
/ ピアノソロを聞きたい時は[a5]を[a8]、[a6]を[r6]に変更し
/ て下さい。
*a*      o5Az95v1318 [a1][a2][a3][a4]
*a1*[!] |:edeg16Bd16^2 Ac>b<c|d16>Bb16^2
         Aa<ceg16g.fef Bg2.r4:| Af16Bd4..>Ab
         <c>b<cd^edc^2.r4
*a2*[!] rcc"1c16d.""cd"edc>b4<dBc^2 Arcccddeff.g4^16rggg
         g4.frfffa4.g4.rc{cdeefg}1f1&f2r4<UUc4>
*a3*[!] |:Ab2r4b4b4a4gf4Bg2..|Ag4g4fe^Bd4.
         |:Ad4eB|e2^:|e4.Ae4q7b2Aag4f2^r4<c4>:|
         efg4ag4b4Be2..Adef4e4de4b&b1&Bb2.r8A{ef}8
*a4*[!] A|:g4aBg4.|Ar{ef}:|g4 <d4c4>gf4Be^2r4Ac4
         q7a2.Aa4g4e4ef4e&
*a5*[!] e1|:5r1:|r3y91,45@30@g12
         r3^3o4v16z110[p1]Arefgq3a.b.<A{ce}
*a6*[!] (fe)&(ef)&f4.ed-c4.(g16a)&a.&(g32a)&a16.g-fg-
         (b12a)&(a12b)&b12^4.ed-c  (d32e)&e4^16.
         DD{d-&c}UU(c32d-)&d-4^16.>D(a32b)&b8..
         {ag-}Ufg-D(a32b)&b16.&(a32b)&b8..U(b32<e)&e16.
         g-aq7bA(a32b)&b4...DD{a&g-}8UU
         r(f32g)&g8..q7bq4b*28ar*20A{ag}
         fe>b<e32&f8..^8e>b<f1&¥80f2..¥0y91,80@74r8[p0]
*a7*[!] e1|:6r1:|r2.z110A<c4>
*a8*[!] e1|:7r1:|
**
/ Piano.
(t20) I[p0]y91,50y93,0v14z95@2o4q818 r4
(t20) [b0][b][b5][b6][b][b7][b3][b4'][b5][b8]
/ ピアノソロを聞きたい時は[b6]を[b6'] に変更して下さい。
/ (通常演奏では[b6'] を打ち込む必要はありません。)
*b0*[!] <rbf<f^e>fg^2^f>ag&g1 <rae<e^>aef^2^e>g-f&f1
         UUUUe*384DD <<e4.d-^2 >e4.d-^4>a4 <e1DDDDf1
*b*      z75o4[b1][b2]z95[b3][b4]
*b1*     [!]12'g<e''f<d''e<c''db''cea''ea<c'
         'dfgb''c4df'>'b%4<dg' [b11][b12]
*b11*    'g<e''f<d''eg<c''fb<d''a<ce'72'b<df'120
*b12*    @d1UU'g4<cf'DD<d8>@d0UU'g<ce%'120
*b2*[!] 12z75|:'a<ce''b<df'|'g4.<cf''g*120<ce':|
         'b4.<df''b*120<eg'<'c4.dg''>b%*120<dg'
         'c4.eg''c*120eg'>'a1<ce' [b21]
*b21*    14'a<ce'<U'ceg'U'cea'U'ea<c'
         U'a<ce'<U'cef'UU",31q4"|:'e8a<c':|r4>
*b3*[!] 11|:'efgb',0'dfb''deb',32|'dfgb',0
         1*120'd4.a<c''ea<c''d4.gb''egb''b2..ceb',24
         14>'a.<cf',0'b<dg'<'cea''ea<c':|'d-*168fb'
         'c.eg''ceg''c2ef'>'b2<f''b.<e'<
         'c.ef''ceg''cea''ea<c''a2<ce''a2<df'
*b4*[!] 14<e*0c.>g8^2<e*0>b%.g8^2<e*0>b.g8^2<e*0>a%.g8^2
         z0<e*0z95[b41]  /ナゼカコノオトガナラナイトキガアル^^^フアンテイ
*b41*    >>12|:'a<ce'|<'ea<c':|U|:'e8a<c':|r4
*b4'*[!]18z100@1<c>ge<g^2 >b%ge<g^2
         >bge<g^2 >a%ge<g^2 @2[b41]
*b5*[!] z9018e4.'g1b'|:{ab}agfe|'d4f'>b<b^2<f4.d^c>b4
         >'b4.<e'<'e1gb':|12>'a<cf''b<dg'<'cea''a<ce'
*b6*[!] 12'b..<d-f''a<ce'216'a1<cef''b<d-f''fb<d-'
         'g-1b<e'UU'b1<eg-'<DD'ceg''ceb'
         >'a4.<cf''a<cf'120'a<cf'384
*b6'*[!]18'b<d-f'120<ed-'c4.>a'a4ag-fg- >a*0<c*0{fe}
         f2edc>z0a*0<z90'c4.e'{dc}'d4.>b'>b^{ag-}
         fg-bb4<e>b*0<e*0g-abb4.&{bag-}4>z0b*0<e*0z90r
         'c4eg'b'bec'ar{ag}'fc>a'e>b<'f4.c>a'e>b'a<cf'384
*b7*[!] 18|:z100'g4.b<e''g4.b'<ec*0r4>>|:'a<ce''a<cf'r
         :|rz0<<c*0:| z10012>>[b71][b71]>
*b71*    'a<ce''a<cf''b<dg'<'cea'
*b8*     z95r8>'a8<ce''a<ce'1080
**
(t21) I[p0]v14z100@2o2q818 r4
(t21) [c0][c]z60[c5][c6][c][c7][c3][c4'][c5][c8]
*c0*[!] 'a<eb'576,24'g-<d-a'576
         z115<'bg-c-'384,0<z105'a>c->f-'384
         z90 'ec>a>b'180 z60 >>'b<a<c-df'204,4
*c*      z80o4q8[c1]z80[c2]z60[c3][c4]
*c1*[!] 12c>bag f>b<ed4>g4 ",60<"c>bag f4.g8^U<c4.c8^
*c2*[!] |:>ab|<c4.c4.>b4:|<e4.e^8 >g4.g^8<c4.>b^8f1
         UUb1&b2UUb8b8r4<
*c3*[!] q718|:|:8e:||:8d:||:8c:||>|:8b:||:8a:|
         |:8g:||:7f:|b&|:8b:|:||:7b:|a&
         |:8a:||:7g:|f&|:8f:||:8b:|
*c4*[!] <|:8c:|>|:8b%:||:8b:||:8a%:|[c41]
*c4'*[!]11",21z80q8<<c>b%ba%>q7z60""rrrrz80"[c41]
*c41*    18|:8f:|bbbbUbbr4
*c5*[!] <|:16e:||:16d:||:16c:|>|:16b:|
*c6*[!] |:16g-:||:8f:||:8b:|<|:16c-:|>",21[c61]""[j61]"
*c61*    |:4'b<b''b<b''b4<b':|A'b<b'384
*j61*    |:4bbb4:|Ab*384
*c7*[!] 18<",21|:|:9e:|UUeereer4:|""|:32e:|">|:32b:|<
*c8*     rUUbb*1080
**
/ Strings,Brass.
(#W,"y91,90@y1,$66,75i0@63") /Syn.Brass1.
(#X,"y91,80@y1,$66,75i8@64") /Syn.Brass4.
(#Y,"y91,50@y1,$66,50i0@50") /Slow Strings.
(#Z,"y91,50@y1,$66,50i0@49") /Strings.
(#V,"u-2:|u+21|:8")
(t30) I[p1]y93,10Xv3z100 o3A116 r4
(t30) [d0][d][d5][d6][d][d7][d3][d4][d5][d8]
*d0*[!] |:5r1:|r*186 [d01][d01] Aa1&_30a*198 r1r1
*d01*    ba~5bg-~ab<~c-d-~ ed~ec~de~fg-~
*d*      "1[r1]""[d1]"[r2]v13[d3][d4]
*d1*[!] Yz90o314A@k8|:3r1:|y6,60g<z80@49gfg
         e<c>dbz60cz80<c>bde.f8^2f.¥30e%8^2@k0¥0
*d3*[!] 14Y@u50Ao4|:b1&b1&¥24b1&b1¥0"3@v113""v13"||:r1:|
         u40i8@49r2..f4.ga<c>i0@50u10:|
         Zz70g2f2<c2>b.z100i8@49f.ga<ce2f2
*d4*[!] |:5r1:|i0@62r2z100o4q7|:'e8a<c':|r4
*d5*[!] r3Wy93,60r3^3z90o4|:5r1:|
         116fcfa<fa<c>c >gdgb<gb<d>d|:c>a<cea<ce|>e:|<c
*d6*[!] d-*384r1r4.z120>>d-cd-2>_8b1&~~~b1__z9012fa<ce
         f*384y93,10
*d7*[!] W18z100o4|:r1|:r'ce''cf':|r4:|@57
         116z85q7|:8eVfVgVau-2:|<z90|:8VeVfVgVau-2:|~10
*d8*[!] y93,1018@49ro4z80q2'ce'¥13A'ce'1080
**
(t31) I[p1]Yv3z100 o3A116 r4
(t31) [e0][e][e5][e6][e][e7][e3][e4][e5][e8]
*e0*[!] |:6r1:|'b<e'384<'ea'384v7'c1e'v5'd1f'
*e*      @v45"1~0Y""~3Z"z100o4[e1][e2][e3][e4]
*e1*[!] 12'ceg'~>'b<df''a<ce''gb<d'~'f<ce''<cea'
         'b<dfg''f4<cd''g4b%<d' [b11] [e12]
*e12*    'g4<cf'72"2¥30"'g<ce%'120"2¥0"
*e2*[!] |:7r1:|Xv13z70o3q7[b21]
*e3*[!] A11z80o4|:'eg''df''ce'>|'b<f''a4.<c'
         'a<c'120'g4.b''gb'120'f2..<e''b<f'216:|'b2..<d-'
         'a<c'216 'g2<f''g4.<e''f<c'216'b2<c''b2<d'
*e4*[!] z9011"3|:4r1:|""o4'cg'>'b%<g''b<g''a%<g'"
         'a<c''a2<c'z75'a8<c''a4.<c'
*e5*[!] z80<'eg'384'df'384'cg'38412>'b<c''b<d''b<e''b<a'
*e6*[!] 11'g-b''g-a''fa'|:'b2<d-':|<'c-eg-'384
         12>|:4'b<ca':|'b<ca'384
*e7*[!] <18|:z90'eg'216UU'ea'q7'e4a'A'e8a'q7'e4a'r:|
         12ADD>'b<ce''b<cf''b<dg''b<ea'
         'b<a<ce''b<a<cf''b<b<dg''b<<cea'~10
*e8*     r8>q2'b8<a<ce'¥13A'b<a<ce'1080
```

```
**

/Guiter,etc...
(t40) I[p1]n4y91,80y63,0v16i0@57
(t40) n9y91,50y63,0v12i0@46z70o5q318 r4
(t40) [r0][f][f5][f6][f][f7][f3][f4][f5]
*f*      l8[r1][r2][f3][f4]
*f3*[!] i0@46o5z80q3|:begebege|:bdfdbdfd|bdedbded:||
         cadce<c>ae >b<gd>b<ebge
         >b<fe>b<cbef >b<fd>b<dbfd:|
         i8@108>z80rggrffr4.<z90ccr>bbr4r1r1
*f4*[!] i0@28z70o4r1r1r1r1c2^8<z90c4>z70c>b2<z80bbr4
*f5*[!] z80@46o5|:|:begbebge:|||:bdfbdbfd:|:|r1r1
*f6*[!] @h48n4@m50o5Az90f2..e^1f1d-2d-2e*384@m0n9v16
         r3y91,80@10r3^3r1o7z80¥40[f61]v12y91,50
*f61*    l8age>bage>ba1¥0
*f7*[!] i0@28v13z120o6[f71]|:4r1:|~5
*f71*    l8|:e4.>b4.<ec^1:|
**

/ Brightness,etc...
(t50) I[p1]n7y91,70y93,60n3y91,70y93,60v8z95@101l8 r4
(t50) [h0][h][h5][h6][h][h7]@v68[h3]n3@101@v68[r4][h5]
*h0*[!] <rbf<f^e>fg^2^>r4.r1 <rae<e^>aef^2^>r4.r1
         r1r1 <<e4.d-^2 >e4.d-^4>a4 <e1DDDDf1n7
*h*      l8o5[r1][h2][h3][h4]
*h2*[!] v16z60@90q8|:[h21]:|r1r1[h21] r3^3@101v9r3
*h21*    ur2|:4u+15{b<fb}8:|>|:8u-9{b<fb}8:|>
*h3*[!] z90o4q7r1r1r1r1l*120 c4.c>b4.br2.. l4<<f.ga<c
         |:5r1:|r2..>f.ga<ce2f2
*h4*[!] o6l*120c4.>gb%4.gb4.ga%4.gr1r1
*h5*[!] n3l8o5|:e4.b1{ab}agfe|f4bb^2r1:|l2fga<e
*h6*[!] d-1|:7r1:|n7z90¥33[f61]
*h7*[!] o6z110[f71]n3@57l16v15r1r1
         z80o6q7|:8eVfVgVau-2:|n7
**

(t51) I[p1]y91,70y93,60v7z80
(t51) @101@y1,$63,90@y1,$66,70o4l8l8 r4
(t51) [i0][i][i5][i6][i][i7][i3][r4][i5]
*i0*[!] r1r2..g2..r4 r1r2..f2..r4
         r1r1r1r1 z85@y1,$63,84<e2.r4 z70f2.r4
*i*      l8[r1][i2][i3][r4]
*i2*[!] |:8r1:|r3i0@101@y1,$63,90@y1,$21,100v13r3^3
*i3*[!] z80o4 |:r1r1d6e6z40b6^2&b4.r2^8|:4r1:|:|
*i5*[!] r3^3@v106z40@90@y1,$63,50@y1,$21,50q8o5r3r1r1
         l16|:f>bu5<dfu5a<cu5>b>bu5:|<fu-15>bu-15<dfu-15a
         <cu-15>b>br2r1r1r3@y1,$63,50@y1,$66,50i8@49r3^3
*i6*[!] o4z80l1d-ccf2f2b*384e2e2e2e2¥40e*384¥0@y1,$66,70
*i7*[!] o4z120|:l8r1^8efrefr4:||:4r1:|~5
**

/Bass.
(t60) I[p1]y91,50y93,0y126,1v14z80@33o3@q5l8 r4
(t60) [j0][j][c5][c6][j]z80[c7]z80[c3][c4'][c5][c8]
*j0*[!] n6|:6r1:|v3|:16r8~5:|r1¥20r1¥0n1|:r1:|
*j*      o3"1[r1]""@q5z75[c1]"o3l2z80[c2]z80[c3][c4]
**

/ Drums.
/Drum.
(t1) @1v16z60o2@r1l8 r4
(t1) [r0][k][k5][k6][k][k7][k3'][k4'][k5']
*k*      l8[r1][k2][k3][k4]
*k2*[!] |:8c1:|r2z100'cd''cd'r4
*k3*[!] z70,90|:15c4d4:|cz80"1|:3u8d:|u""dUUdUUU{dd}"
         z70,90|:14c4d4:|z60cz99aaaaz90{gg}f"1c""a"
*k3'*[!]|:z70,100|:14c4d4:||cz100aaaa.f16z110{dd}f
         :|z100faaaa.g16fz110d
*k4*[!] |:4z70c4|c4c4z110d64d...:|z70cz110
         d64d8...z80"1ag{ff}""fz110{dd}z80f"
         z70,90|:3c4d4:|z70cc"1z90d4""z110d64d8..."
*k4'*[!]r1r1r1r4.z100b64b8...bf4
         z70,100|:3c4d4:|z70ccz110d64d8...
*k5*[!] z70,90|:14c4d4:|cz70|:3du17:|
         z110dz80,100{cd}cz110{dd}
*k5'*[!]z70,100|:15c4d4:|z80cz100az110{dd}z70cz100rdd
*k6*[!] z70,90|:11c4d4:|z90{aa}aDDgDD{ff}
         z70,85,80|:4cbf4:|c*384
*k7*[!] |:z70,90|:c4d4:|c*0|:rz100dUd:|z70cz110{dd}:|
         z70,90|:6c4d4:|z70cz110aaaaa{dd}z100f
**

/Symbal,Hihat.
(#H,"o2g-") /Closed Hi-hat.
(#P,"o2a-") /Pedal  Hi-hat.
(#O,"o2b-") /Open   Hi-hat.
(#R,"o3b")  /Ride  Symbal.
(#C,"o3d-") /Crash Symbal.
(t2) v16z55@r1l8 r4
(t2) [r0][l][l5][l6][l][l7][l3][l4'][l5']
*l*      l8[l1][l2][l3][l4]
*l1*[!] |:7r1:|r4."2z30C4""r4" z55{RURUR}8 UR4
*l2*[!] |:7r2H2:|r4|:u+5P4:|z45|:3u+20O4:|r2
*l3*[!] "1[l31]""[l32]"
*l31*    z70C*0z70,70,50,70|:14HHOH:|z70HPPz95Oz80
         |:4u5H:|z40C*0z70,70,50,70|:12HH|OH:|z40C*0
         z50Oz70Hz70,70,50,70|:2HH|OH:|z70HO|:8u4H:|
*l32*    z80C4|:z50,70|:25OH:||z70HOHz50
         "2O^1""O4z60HHz80Oz50H2"z55C4:|z50Oz70HHz80O^1
*l4*[!] z80o3e-*0z70,60|:14PP:|r2z60C4
         z70,60|:5PP:|"1rz60,70|:P|P:|""r2"
*l4'*[!]|:4r1:|z60C4 |:3z50Oz70P|PP:|r2
*l5*[!] z50,70|:16OP:|C4z50,70|:10OP:|z80,100POPr2..
*l5'*[!]z50,70|:16OP:|C4z50,70|:9OP:|z50|:5O4u10:|Pz60O
*l6*[!] z60C4z50,70|:15OP:|z55|:3Ou20Pu-15:|z70O4z75Oz90
         PO2.z100|:5C2:|r2r1
*l7*[!] z70,60|:5P|P:|r2..|:4P|P:|z80OPr2..
         z70,60|:5PP:|z50|:6O4u10:|Pz127O^1
**

/etc...
(t3) v16z40@r1@y29,31,127l8 r4
(t3) [r0][m][m5][m6][m][r7][m3][r4][m5]
*m*      l8[r1][m2][m3][r4]
*m2*[!] o1|:4rz40@y24,31,64@y28,31,1  u|:4 gu-5:|
               z30@y24,31,74@y28,31,127u|:11gu-3:|:|r1
*m3*[!] r3@y$1c,82,1r3^3z70|:8[m31]|r1:|
*m31*    l16r2^8o5y6,1 b-b- y6,127 b-b-b-r
*m5*[!] |:4r1[m31]:|
*m6*     [m5]r1r1
**

*r0*[!] |:12r1:|
*r*      [r1][r2][r3][r4]
*r1*[!] |:8r1:|
*r2*[!] |:9r1:|
*r3*[!] |:16r1:|
*r4*[!] |:6r1:|
*r5*     [r1]
*r6*[!] |:10r1:|
*r7*     [r1]
**
#ext_off /MML拡張展開を終わります。
(p)
/ step count (ループカウントは省略)
/   1:00006DF8    2:00006DB0    3:00006DB0   10:00006DB0
/  20:00007218   21:00007218   30:00007218   31:00007218
/  40:00006DB0   50:00006DB0   51:00006DB0   60:00007218
```

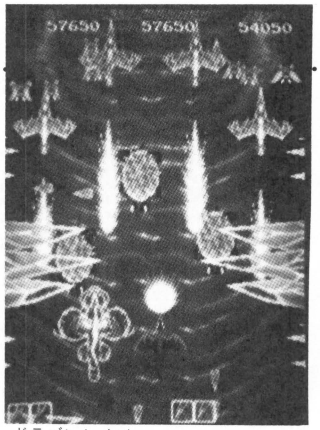
X68000・Z-MUSIC用（SC-55対応）

ドラゴンセイバーより

エンディングテーマ

©NAMCO ALL RIGHTS RESERVED

Nishimoto Hideki
西本 英樹



ドラゴンセイバー

お馴染みドラゴンセイバー

それでは常連さんもいってみましょうか。西本英樹君の作品はナムコのシューティングゲーム「ドラゴンセイバー」のエンディングテーマです。こちらもZ-MUSIC+SC-55が必要になります。

「ドラゴンセイバー」はあの「ドラゴンスピリット」の続編（？）としてこの世に生を享けたファンタジーシューティングゲームです。「ドラゴンスピリット」はX68000用も発売されていますし，かなりヒットした作品ですから，みなさんご存じですよね。

なぜか昔から西本君はエンディングテーマに縁があるようで，タイトーの「NINJA WARRIORS」から「PARADOX」，セガの「TURBO OUTRUN」から「チェッカーフラッグ」，そして今回はナムコ，とメーカーにはこだわらないわりにはエンディングテーマにこだわるという感じですね。実際にはほかにも送られてきているのですが，やっぱり常連にもボツはあるもの。結局，採用された曲にエンディングテーマが多いだけなんですけどね。

コメント中に「もうお忘れでしょうね」などとありましたが，そんなことはありません。常連さんとして，これからも投稿を続けてくださいね。

リスト7

```
==================  ds_ed.zms  ==================
/
/          ドラゴンセイバー <エンディング>
/
/       Composed by 細江 慎治 / (c) namco
/
/            Program arranged by 西本 英樹
/
/       for X68000 Z-MUSIC
/          SC-55/155/CM-300 (GS STANDARD)
/
/          ZMUSIC -T32 でOK
(i)
(b0)

/- ( Track Assign ) -----------------------------------

(m1 ,2000)(aFM1,1)
(m2 ,2000)(aFM2,2)
(m3 ,2000)(aFM3,3)
(m4 ,2000)(aFM4,4)
(m5 ,2000)(aFM5,5)
(m6 ,2000)(aFM6,6)
(m7 ,2000)(aFM7,7)
(m8 ,2000)(aFM8,8)

(m10,2000)(aMIDI1,10)
(m11,2000)(aMIDI2,11)
(m12,2000)(aMIDI3,12)
(m13,2000)(aMIDI4,13)
(m14,2000)(aMIDI5,14)
(m15,2000)(aMIDI6,15)
(m16,2000)(aMIDI7,16)

(m17,2000)(aMIDI10,17)
(m18,2000)(aMIDI10,18)
(m19,2000)(aMIDI10,19)
(m20,2000)(aMIDI10,20)

(o71)
.roland_exclusive $10,$42 ={$40,00,$7F,00} /Reset

/- ( FM Voice ) ---------------------------------------

/                                                 Bass
/            AR D1R D2R  RR D1L  TL  KS MUL DT1 DT2 AMSE
(@1          26,  1,  0,  2,  2, 28,  1,  1,  3,  0,  0,
             31,  2,  0,  6,  2,  0,  1,  4,  3,  0,  0,
             28,  1,  0,  2,  2, 40,  1,  1,  7,  0,  0,
             31,  2,  0,  6,  2,  0,  1,  1,  7,  0,  0,
/           ALG  FB  OM PAN
              4,  0, 15,  3)
/                                                 Bell
/            AR D1R D2R  RR D1L  TL  KS MUL DT1 DT2 AMSE
(@2          31, 12,  5,  5, 15, 24,  2,  5,  3,  0,  0,
             31,  9,  8,  6, 15,  2,  1,  2,  3,  0,  0,
             31,  9,  8,  6,  6,  9,  1,  1,  7,  0,  0,
             31,  8, 10,  6,  4, 10,  1,  2,  7,  0,  0,
/           ALG  FB  OM PAN
              6,  7, 15,  3)
/                                               Faid in Cho
rus
/           AR D1R D2R  RR D1L  TL  KS MUL DT1 DT2 AMSE
(@3           3,  2,  0,  2,  2, 36,  0,  8,  3,  0,  0,
              4,  6,  0,  5,  9,  0,  0,  8,  3,  0,  0,
              3,  2,  0,  2,  2, 36,  0,  8,  7,  0,  0,
              4,  6,  0,  5,  9,  0,  0,  8,  7,  0,  0,
/           ALG  FB  OM PAN
              4,  0, 15,  3)
/                                              Strings
/          AR D1R D2R  RR D1L  TL  KS MUL DT1 DT2 AMSE
(@4        16,  1,  0,  2,  2, 30,  1,  4,  7,  0,  0
           19,  1,  0,  2,  2, 36,  1,  8,  0,  0,  0
           18,  1,  0,  2,  2, 37,  1, 12,  2,  0,  0
           14,  2,  0,  7,  2,  8,  1,  8,  0,  0,  0
/         ALG  FB  OM PAN
            2,  7, 15,  3)

/- ( Music Data ) -------------------------------------
/
/           FM Chord
/
(t1)    N1  @03 @U127 V11 O1 Q8 L8  K0 @K2
(t1)    @M1 @H1 @S1 r4 P3
(t1)    |:2 e-*336 rr>b-*336<rr
(t1)    e-2.rr e-2.rr b-2.rrb-2.rr :|
(t1)    @01 O1 V13 @K0
(t1)    |:2 g1g1f1f1 :|
/
/           FM Chord
/
(t2)    N2  @03 @U127 V11 O1 Q8 L8  K0 @K0
(t2)    @M1 @H1 @S1 r4 P3
(t2)    |:2 c*336rr>g*336rr<
(t2)    c2.rrc2.rr>g2.rrg2.rr :|
(t2)    @01 O1 V11 @K-2 P1 r16
(t2)    |:2 g1g1f1|f1 :| f*180
/
/           FM Chord
/
(t3)    N3  @03 @U127 V11 O1 Q8 L8  K0 @K4
(t3)    @M1 @H1 @S1 r4 P3
(t3)    |:2 b-*336 rr f*336 rr
(t3)    b-2.rr b-2.rr f2.rrf2.rr :|
(t3)    @01 O1 V10 @K3  P2 r16.
(t3)    |:2 g1g1f1|f1 :| f*174
/
/           FM Chord
/
(t4)    N4  @03 @U127 V11 O1 Q8 L8  K0 @K-2
(t4)    @M1 @H1 @S1 r4 P3
(t4)    |:2 g*336rr d*336rr
(t4)    g2.rrg2.rr d2.rrd2.rr :|
(t4)    @01 O1 V10 @K6  P3 r8
(t4)    |:2 g1g1f1|f1 :| f2..
/
/           FM Bell
/
(t5)    N5  @02 @U127 V10 O5 Q8 L16 K0 @K0
(t5)    @M1 @H1 @S1 P3
(t5)    r4 r*1536
(t5)    |:2 b-2.a4 g1 d*384 :|
(t5)    |:2 g4.c3... b-4f4g4e-4
(t5)    |1  d2 >b-*288< :| f1a1
/
/           FM Bell
/
(t6)    N6  @02 @U127 V06 O5 Q8 L16 K0 @K4
(t6)    @M1 @H1 @S1.P3 r8
(t6)    r4 r*1536
(t6)    |:2 b-2.a4 g1 d*384 :|
(t6)    |:2 g4.c3... b-4f4g4e-4
(t6)    |1  d2 >b-*288< :| f1a1
/
/           FM Strings
/
(t7)    N7  @04 @U127 V14 O4 Q8 L16 K0 @K-2
(t7)    @M1 @H1 @S1 P3
(t7)    r4 r*1536
(t7)    |:2 b-2.a4 g1 d*384 :|
(t7)    |:2 g4.c3... b-4f4g4e-4
(t7)    |1  d2 >b-*288< :| f1a1
/
/           FM Strings
/
(t8)    N8  @04 @U127 V11 O4 Q8 L16 K0 @K6
(t8)    @M1 @H1 @S1 P3 r8
(t8)    r4 r*1536
(t8)    |:2 b-2.a4 g1 d*384 :|
(t8)    |:2 g4.c3... b-4f4g4e-4
(t8)    |1  d2 >b-*288< :| f1a1
/
/          Arpedio 1
/
(t10)   N11 @47 @U099 V11 O4 Q8 L16 K0 @K0
(t10)   @I$41,1,$42 @E040,060 @D0 @P63
(t10)   @Y1,33,0  @Y1,101,64 @Y1,100,64 r4
/
(t10)   |:4|:2 r>b-<cd b-8e-c8 b-e-f db-e-g :|
(t10)   |:2 r>gab-< f8c>g8<  f>ab- g<f>ab- :|:|<
/
(t10)   |:2 ge-f>g8 <cdg8 >b-<d>b- <b-gad
(t10)       ge-f>g8 <cdg8 >b-<d>b- <b-gab-
(t10)       fcd>a8<  cdf8 >a<c>a <afge-
(t10)       fcd>a8<  cdf>fgab- l32Gab-<cfgab- l16 :|
/
(t10)   r>b-<cd b-e-cf >rgab-< dc>g<d
(t10)   r>b-<cd b-8e-8 >rgab-< dc>g8<
(t10)   r>b-<cd b-e-cf >rgab-< dc>g8<
(t10)   T60>b-<T56cT52dT52e- T48b-<T44cT40dT36e-b-<d1
/
/          Arpedio 1 Delay
/
(t11)   N12 @47 @U099 V08 O4 Q8 L16 K0 @K8
(t11)   @I$41,2,$42 @E040,060 @D0 @P99
(t11)   @Y1,33,0  @Y1,101,64 @Y1,100,64 r*57
/                                       こからトラック10と同じ
(t11)   |:4|:2 r>b-<cd b-8e-c8 b-e-f db-e-g :|
(t11)   |:2 r>gab-< f8c>g8<  f>ab- g<f>ab- :|:|<
/
(t11)   |:2 ge-f>g8 <cdg8 >b-<d>b- <b-gad
(t11)       ge-f>g8 <cdg8 >b-<d>b- <b-gab-
(t11)       fcd>a8<  cdf8 >a<c>a <afge-
(t11)       fcd>a8<  cdf>fgab- l32Gab-<cfgab- l16 :|
/
(t11)   r>b-<cd b-e-cf >rgab-< dc>g<d
(t11)   r>b-<cd b-8e-8 >rgab-< dc>g8<
(t11)   r>b-<cd b-e-cf >rgab-< dc>g8<
(t11)   >b-<cde- b-<cde-b-<d1
/
/           Arpedio 2
/
(t12)   N13 @47 @U099 V12 O2 Q8 L8  K0 @K0
(t12)   @I$41,3,$42 @E040,060 @D0 @P32
(t12)   @Y1,33,64 @Y1,101,64 @Y1,100,64 r4
/
(t12)   |:4|:2 @P32 gb- @P48'<ce-gb-2.' :|
(t12)      |:2 @P30 e-g @P40'gb-<df2.'  :|:|
/
(t12)   @Y1,33,64 @55 @U80 V06 O4 Q2 @E127,127
(t12)   |:8'gb-<df4':||:8'fa<cd4':|
(t12)   |:8'gb-<df4':||:4'fa<cd4':||:4'a<cdf4':|
/
(t12)   @Y1,33,64 @47 @U99 V12 O2 Q8 @E040,060
(t12)   |:3 @P32 gb- @P48'<ce-gb-4'
(t12)       @P30 e-g @P40'gb-<df4' :|
(t12)       @P30 e-g @P40'gb-<df2.'
/
/           Strings
/
(t13)   N14 @50 @U099 V10 O5 Q8 L8  K0 @K0
(t13)   @I$41,4,$42 @E020,127 @D0 @P64
(t13)   r4 r*1536
(t13)   |:2 b-2.a4 g1 d*384 :|
```

```
(t13)   |:2 g4.c3... b-4f4g4e-4
(t13)   |1  d2 >b-*288< :| f1a1
/
/           Bell
/
(t14)   N15 @10 @U127 V12 O5 Q8 L8  K0 @K0
(t14)   @I$41,5,$42 @E127,127 @D0 @P64
/                                       ここからトラック13と同じ
(t14)   r4 r*1536
(t14)   |:2 b-2.a4 g1 d*384 :|
(t14)   |:2 g4.c3... b-4f4g4e-4
(t14)   |1  d2 >b-*288< :| f1a1
/
/           Chorus
/
(t15)   N16 @53 @U099 V08 O4 Q8 L8  K0 @K4
(t15)   @I$41,6,$42 @E127,127 @D0 @P64
(t15)   @M1 @H60 r4
/
(t15)   |:2 |:2 Q8 @Y1,99,100
(t15)   ¥0@u99¥20'e-cb-g'*384 ¥0@u99¥20'>g<fd'*384
(t15)   @Y1,99,90 |:2 ¥0@u99¥40'e-cb-g1' :|
(t15)   |:2 ¥0@u99¥40 '>g<fd1' :| :| ¥0
/
(t15)   @55 O4 V08 @U70 @Y1,99,64 Q8 r*1536
(t15)   |:3 P3V11'ce-gb-4' P1V7'ce-gb-' P2V5'ce-gb-'
(t15)       P3V11'>b-<df4' P2V7'>b-<df' P2V5'>b-<df':|
(t15)   @M1 V11 P3 @D1 'ce-gb-2' ¥80 r1
/
/           Effect & Partial reserve
/
(t16)   N10 @I$41,0,$42
(t16)   X$40,00,00,00,04,00,00
(t16)   X$40,01,$30,03 X$40,01,$38,02
(t16)   X$40,1,$10,6,0,6,1,4,1,4,2,0,0,0,0,0,0,0,0
(t16)   r4
/
/           Drums (Bell / High Hat)
/
(t17)   N19 @01 @U099 V16 O4 Q8 L16 K0 @K0
(t17)   @I$41,9,$42 @E127,127 @R1
(t17)   r4
(t17)   |:24 r4 @U50O5b4
(t17)   @Y28,42,60 @U40O2f+ @Y28,42,32@U45f+ r4. :|
/
/           Drums (Bass Drum)
/
(t18)   N19     @U127     O1 Q8 L16
(t18)   @I$41,9,$42 @E127,127 @R1
(t18)   r4 r*1536
(t18)   |:8 @U118 b4r8.@U80b @U70b4. @U127b8 :|
/
/           Drums (High Hat)
/
(t19)   N19     @U127     O2 Q8 L16
(t19)   @I$41,9,$42 @E127,127 @R1
(t19)   r4
(t19)   |:2 r*768 |:4 @U40a+8.@U1f+*156 :| :|
/
/           Drums (Cymbal)
/
(t20)   N19     @U127     O2 Q8 L16
(t20)   @I$41,9,$42 @E127,127 @R1
(t20)   r4 r*3072
(t20)   'c<g1'r*576 <g1>r*576
(p)
/-----------------------------------------------------
/         肩のコらない短かさが今回のウリです。
/         作った本人は首が回らない程、肩コリに
/       悩みましたが…(泣)。
/         次回もよろしくっ!!
/                                 西 本 英 樹 (Sun West)
/-----------------------------------------------------
```

リスト8 トータルカウント（参考）

```
DRAGON SABER [ Ending ] Programed by Hide cha!

- Counter -
 1:00001230 00000000  2:00001230 00000000  3:00001230 00000000  4:00001230 00000000
 5:00001230 00000000  6:00001248 00000000  7:00001230 00000000  8:00001248 00000000
10:0000159C 00000000 11:000015A5 00000000 12:00001530 00000000 13:00001230 00000000
14:00001230 00000000 15:00001590 00000000 16:00000030 00000000 17:00001230 00000000
18:00000C30 00000000 19:00000C30 00000000 20:00001230 00000000
```

X68000・Z-MUSIC用

Weapon・Shop

（オリジナル曲）

Yabe Masatoshi
矢部 雅敏

武器屋のテーマはブキミ？

さらにオリジナル曲も掲載しましょう。このコーナーでもコンポーザーとしてかなり定着してきた感のある矢部君の登場です。この曲はZ-MUSIC用ですが，X68000本体だけで鳴らすことができますので，MIDIを持っていない人でもちゃんと聴けますよ。

曲のタイトルは「Weapon・Shop」です。その名のとおり，武器屋のBGMとして使うことを前提に作曲されたそうです。この曲を使うゲーム自体は，まだ影も形もないらしいのですが。

曲の雰囲気はファルコムから出ていた「ソーサリアン」の「ペンタウェア」に似ていなくもない感じですが，かっこよくまとまっています。小見出しにブキミなどと書いてしまいましたが，あれはシャレです。勘弁してください。

昔の作品に比べて，着実に力を伸ばしているのがわかります。このままもっと素晴らしい曲を作りつづけてください。

矢部君はかなりのハイペースで投稿しつづけており，「戦いの兜」以来すっかり常連さんとなっています。誌上でも常連さんを目指してがんばってくださいね。

ついでに，現在Oh!Xではオリジナル曲による投稿を大々的に募集しています。6月号の付録ディスクでオリジナル曲を収録したためか，最近はオリジナル曲の投稿が少しずつ増えてきました。データがたまればオリジナルデータ集としてまとめる予定もあります。皆さんのご参加をお待ちしています。

リスト9

```
==================  WEAPON.ZMS  ==================
.comment  - WEAPON・SHOP -   Compose & Programed  by Yabe
/-----------------------------------
/
/        for  ZMUSIC.X  Ver 1.10
/
(i)
(b0)
(m1,2000)
(m2,2000)
(m3,2000)
(m4,2000)
(m5,2000)
(m6,2000)
(m7,2000)
(m8,2000)
(m9,2000)
(a1,1)
(a2,2)
(a3,3)
(a4,4)
(a5,5)
(a6,6)
```

```
(a7,7)
(a8,8)
(a9,9)


/--------------------------------------------------
.ADPCM_BLOCK_DATA=WEAPON.ZPD


/--------------------------------------------------

(v1,0
/          AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
           61, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/          AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
           31, 11,  0,  0,  1,  7,  0,  0,  0,  0,  0
           23, 13, 12,  7,  2, 20,  0,  6,  3,  0,  0
           31, 13, 12,  7,  1, 20,  0,  6,  7,  0,  0
           31, 17, 12,  7, 13,  2,  0,  6,  0,  0,  0)
(v2,0
/          AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
           59, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/          AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
           31, 23, 15,  2,  3,  8,  0, 14,  0,  0,  0
           31, 20, 15,  2,  3, 34,  0,  8,  0,  0,  0
           31, 18, 15,  2,  3, 30,  0,  7,  1,  0,  0
           31, 18, 15,  8,  3,  0,  0,  2,  0,  0,  0)
(v3,0
/          AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
            3, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/          AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
           31,  6,  3,  3,  2, 39,  1,  9,  3,  0,  0
           27,  6,  1,  4,  1, 43,  1,  6,  0,  0,  0
           26,  8,  1,  5,  2, 44,  1,  0,  0,  0,  0
           28,  7,  1,  8,  6,  0,  1,  1,  7,  0,  0)
(v4,0
/          AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
           32, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/          AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
           31, 14,  8,  6,  5, 38,  2, 12,  3,  0,  0
           25, 10,  7,  5,  4, 45,  2,  2,  0,  0,  0
           28,  8,  3,  6,  1, 22,  1,  0,  0,  0,  0
           25,  7,  1,  6,  3,  0,  1,  0,  7,  0,  0)
(v5,0
/          AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
           61, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/          AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
           31, 10,  5,  6,  5, 32,  1,  2,  0,  0,  0
           31,  8,  7,  5,  4,  5,  1,  4,  3,  0,  0
           31,  8,  3,  6,  1,  8,  0,  6,  5,  0,  0
           31,  7,  1,  7,  3,  0,  0,  1,  7,  0,  0)


/--------------------------------------
(t1)r1[do]|: @1o5l16v14
(t1)|:r1r2p1c8.p2c8.rrr1r1:|  :|
/--------------------------------------
(t2)r1[do]    @4o2l16 v14
(t2)|:g8.grb-8.fg8<c8.>b-8
(t2)  g8.grb-8.fg8|1<d8.c8>:|f8q2dq8d8
(t2)|:g8.grb-8.fg8<c8.>b-8
(t2)  g8.grb-8.|1fg8<d8.c8>:|<ddddrrrr [loop]
/--------------------------------------
(t3)r1r32[do]   @4o2l16 v11  @k7
(t3)|:g8.grb-8.fg8<c8.>b-8
(t3)  g8.grb-8.fg8|1<d8.c8>:|f8.d8
(t3)|:g8.grb-8.fg8<c8.>b-8
(t3)  g8.grb-8.|1fg8<d8.c8>:|<ddddrrrr  [loop]
/--------------------------------------
(t4)r1[do] @3o4v14 l16        @h24 @s6 @m6
(t4)gda2.b-4.g8df4.. f+32&g2&g8..<_8|:8f64~1:|f4 f+32g8&
g32 drr>b-2&b-8
(t4)gda2.b-4.g8de48&f4.&f24<d2. g4.g+32a8&a32<crr>b-2&b-
8 [loop]
/--------------------------------------
(t5)r1r32.[do]  @3 o4 v11 l16   @k4 @h24 @s6 @m6
(t5)gda2.b-4.g8df4.. f+32&g2&g8..<_8|:8f64~1:|f4 f+32g8&
g32 drr>b-2&b-8
(t5)gda2.b-4.g8de48&f4.&f24<d2. g4.g+32a8&a32<crr>b-2&b-
8  [loop]
/--------------------------------------
(t6)r1[do]  @5o3l16v13
(t6)r1r2rrdr rrrrr1r2rrfr rrrr
(t6)r1r2rrdr r4  r1r2dddd rrrr [loop]
/--------------------------------------
(t7)r1[do]  @5o3l16v13    p1
(t7)r1r2rrfr rrrrr1r2rrar rrrr
(t7)r1r2rrfr r4  r1r2ffff rrrr [loop]
/--------------------------------------
(t8)@2v13 c4c4c4r4[do]@5o4l16v13    p2
(t8)r1r2rr>ar<rrrrr1r2rrcr rrrr
(t8)r1r2rr>ar<r4 r1r2dddd rrrr  [loop]
/--------------------------------------
(t9)o4l16  e4e4e4o2c4[do] p3 @f4
(t9)|:15o1c4<c4> :|<cccc r@f2o2p1cp2c8>>
(t9)[loop]
(p)
```

リスト10 Weapon･Shop用コンフィグファイル

```
1=SNARE¥FTAS1.PCM,,V45
.O1C = BASS¥CTRLK.PCM,,V80
.O2C =SNARE¥RVBS1_.PCM,,V77,M1
.O4E =CYMBAL¥HH1.PCM,,V80

.ERASE 1
```

リスト11 トータルカウント（参考）

```
/---- WEAPON･SHOP
 1:000000CC0 00000000   2:000000C0 00000600   3:000000C6 00000600   4:000000C0 00000600
 5:000000C9 00000600   6:000000C0 00000600   7:000000C0 00000600   8:000000C0 00000600
 9:000000C0 00000600
```

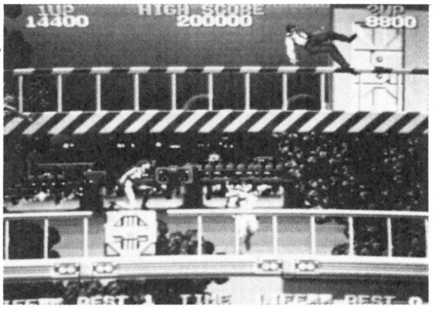
X68000・Z-MUSIC＋PCM8用（CM-64＋SN-U110-07対応）

ローリングサンダー2より

©NAMCO ALL RIGHTS RESERVED

TERROR FOR SALE

Shindo Noriyuki
進藤 慶到



ローリングサンダー2

要CM-64＋……

皆さん，こんにちは。今回はナムコのアクションゲーム，ローリングサンダー2より「TERROR FOR SALE」をお送りします。ただ，演奏にはZ-MUSICシステムはもちろんのこと，PCM8.XとCM-64，またさらにギターカード(SN-U110-07)が必要でして，いったい何人の方に聴いてもらえるのか，少々不安でもあります。まあ，ギターカードを持ってなくても，リストの注意書きに従って変更を加えれば一応演奏できますので，ぜひ聴いてください。

このゲームの曲は，なんというか独特の雰囲気を持っているんですよね。切れのあるブラス隊と，ちょっと懐かしい感じが漂うエレクトリックオルガンとの絡みなんかすごく渋いんですよ。また，ほぼ全体を通して鳴り響いているシンセタムっぽい音や，ゲートの効いたドラムス，オシャレなラテンパーカッションなど，リズムも面白いんです。イメージとしてはジャズ風なのにまったく古さを感じませんし，ちょっと変わった曲なのですが一度聴いただけで気にい

ってしまいました。善バビの西川氏もお気にいりだそうですよ。

このプログラムは，そんなに凝ったテクニックは使っていません。FM音源は少ししか鳴ってませんし，MIDIも当たり前の使い方しかしていません。ただ，曲自体が長いので，それに応じてリストも長くなってしまいました(いつものことかな)。重複部分もかなりあります。コーラスやエコーなどを同じ音を重ねて再現しているからです。リストをよく見て，うまく入力してください。

さて，この曲，できるだけ本物に似せようと努力したんですが，ドラムだけは無理でした。X68000のAD PCMはステレオ発音ではないので，どうしても真ん中にこもってしまうんです。複数の音源をお持ちの方は，ドラムもMIDI対応にしてみてください。迫力が出ますよ。SC-55のパワーセットが手軽でいいと思います。音もそっくりですしね。

では，またの機会にお会いしましょう。感想やリクエストなどがあればぜひ送ってくださいね。　　　　(進藤慶到)

リスト12

```
.comment -ROLLING THUNDER2- TERROR FOR SALE (C)namco  by EN
G 92/06/28 (+PCM8,CM64&07)
/ for ZMUSIC.X + PCM8.X
/ MIDI MODULE : CM-64 + SN-U110-07
/------------------------------------------------------------
/ TRACK SETUP
(i)
/ INTERNAL
(m1,1000)(aFm1,1)
(m2,1000)(aFm2,2)
(m3,1000)(aFm3,3)
(m4,1000)(aFm4,4)
(m5,1000)(aFm5,5)
(m6,1000)(aFm6,6)
(m7,1000)(aFm7,7)
(m8,1000)(aFm8,8)
(m9,1000)(aAdpcm,9)
(m10,1000)(aAdpcm,10)
/ CM64
(m11,1000)(aMidi6,11)
(m12,1000)(aMidi7,12)
(m13,1000)(aMidi8,13)
(m14,1000)(aMidi9,14)
(m15,1000)(aMidi10,15)
(m16,1000)(aMidi11,16)
(m17,1000)(aMidi12,17)
(m18,1000)(aMidi5,18)
(m19,1000)(aMidi1,19)
(m20,1000)(aMidi1,20)
(m21,1000)(aMidi2,21)
(m22,1000)(aMidi3,22)
(m23,1000)(aMidi4,23)
/------------------------------------------------------------
/ CM64 INIT
.roland_exclusive 16,22={$7F,00,00,00}
/------------------------------------------------------------
/ ADPCM DATA SET
.adpcm_block_data=TERROR.ZPD
/------------------------------------------------------------
/ OPM DATA SET
/        AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2 AME  BRASS
(@1,     19, 31,  0,  9,  0, 30,  0,  1,  0,  0,  0
         19, 14,  0,  9,  1, 65,  0,  5,  5,  3,  0
         19,  0,  0,  9,  0, 34,  0,  2,  0,  0,  0
         14,  0,  0, 11,  0,  3,  0,  3,  1,  0,  0
/        AL  FB  OM PAN
          2,  7, 15)
/        AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2 AME  NOISE
(@2,      0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0
          0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0
          0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0
         31,  0, 19,  7,  0,  3,  1,  0,  0,  0,  0
/        AL  FB  OM PAN
          0,  0,  8)
/        AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2 AME  CHORD 1
(@3,     31, 21,  0, 15,  1, 23,  0,  1,  0,  0,  0
         31,  4,  0, 15,  7, 14,  0,  1,  0,  0,  0
         31,  4,  0, 15,  7, 14,  0,  1,  0,  0,  0
         31,  4,  0, 15,  7, 14,  0,  2,  0,  0,  0
/        AL  FB  OM PAN
          5,  7, 15)
/        AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2 AME  CHORD 2
(@4,     31, 21,  0, 15,  1, 22,  0,  1,  0,  0,  0
         31,  6,  0, 15,  1, 14,  0,  1,  0,  0,  0
         31,  6,  0, 15,  1, 14,  0,  1,  0,  0,  0
         31,  6,  0, 15,  1, 14,  0,  2,  0,  0,  0
/        AL  FB  OM PAN
          5,  7, 15)
/------------------------------------------------------------
/ LA DATA SET
/        Chord (Saw&Squ)
.roland_exclusive 16,22 ={8,0,0
          67, 72, 79, 82, 68, 32, 32, 32, 32, 32
           0,  0,  3,  0
/ ** PARTIAL1 **
          36, 53, 11,  1,  2,  0, 36,  9
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
          62, 23, 85
         100, 23,  4, 39,  8, 24
         100,  0,  0,  0, 13, 24, 77, 83,100, 89, 76, 72
          94, 74, 91, 12, 27, 12
           0,  0,  0,  0,  0,  9, 43,100,100,100, 99
/ ** PARTIAL2 **
          36, 48, 11,  1,  3,  0,  0,  0
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
          64, 23, 56
         100, 23,  5,103,  6, 43
         100,  0,  0,  0,  7, 22,100, 98,100, 94, 84, 79
          94, 70, 91, 12, 27, 12
           0,  0,  0,  4, 11, 14, 15,100, 96, 92, 91}
/----------------------------
/        Synth Tom
.roland_exclusive 16,22 ={8,2,0
          82, 95, 84, 79, 77, 84, 79, 77, 32, 32
           2,  0,  3,  0
/ ** PARTIAL1 **
          57, 49, 11,  1,  0,  2,  3, 13
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
           0,  0,  0
         100,  0,  5,103,  6,  0
           0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0
          78, 50, 91, 12, 27, 12
           0,  0,  0,  4, 11, 14, 15,100, 96, 92, 91
/ ** PARTIAL2 **
          12, 52, 11,  1,  0,  0, 36,  8
          10,  0,  4, 48,  0,  0,  0, 97,  0,  0,  0,  0
           0,  0,  0
          80, 22,  4, 39,  8, 24
         100,  0,  0,  0, 10, 10, 50,  0,100, 53, 40,  0
         100, 74, 91, 12, 27, 12
           0,  0,  0,  9,  0, 68,  0,100,100, 80,  0}
/----------------------------
/        Keyboard (Harpsichord)
.roland_exclusive 16,22 ={8,4,0
          72, 65, 82, 80, 83, 73, 67, 79, 82, 68
           2,  5,  3,  0
/ ** PARTIAL1 **
          36, 46, 16,  1,  0, 43,  0,  7
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
           0,  0,  0
           0,  0,  3,  0,  7,  0
           0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0
          87, 77, 91,  6, 27, 12
           3,  0,  0, 23, 60, 83,  0,100, 88, 88, 88
/ ** PARTIAL2 **
          48, 51, 16,  1,  2,  0,100,  7
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
           0,  0,  0
         100, 16,  9,103,  8,  0
           0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0
         100, 75, 27, 12, 15,  5
           1,  1,  0, 23, 60, 92,  0,100, 88, 88, 88}
/----------------------------
/ PATCH PARAMETERS SET
/        @1 : Chord
/        @2 : Tom
/        @3 : Keyboard
.mt32_patch 1,16 ={2,0,24,52,2,0,1}
.mt32_patch 2,16 ={2,1,24,50,2,1,1}
.mt32_patch 3,16 ={2,2,24,50,2,0,1}
/------------------------------------------------------------
/ CM64 System SETUP
/ LA SOUND PART
.roland_exclusive 16,22 ={
        $10, 0, 0                           / address
         64                                 / master tune
          2, 4, 5                           / reverb
          6, 6, 6, 6, 4, 0, 0, 0, 4         / ptl reserve
          0, 1, 2, 3, 4,16,16,16, 5         / MIDI ch#
         57}                                / master volume
/ PCM SOUND PART
.roland_exclusive 16,22 ={
        $52, 0, 0                           / address
         64                                 / master tune
          2, 4, 4                           / reverb
```

```
                4, 5, 6, 8, 4, 4                    / ptl reserve
                6, 7, 8, 9,10,11                    / MIDI ch#
               60}                                  / master volume

/-----------------------------------------------------------------
/ DRUM SETUP

.mt32_drum_setup 41,16 ={70,63,11,1}     / Closed Hihat
.mt32_drum_setup 43,16 ={71,63,11,1}     / Open Hihat
.mt32_drum_setup 45,16 ={72,78, 5,1}     / Crash cymbal 1
.mt32_drum_setup 46,16 ={72,78,11,1}     / Crash cymbal 2

/-----------------------------------------------------------------
/ MML DATA SET

(o160)

(t1)    r6 @1 o4 v14 p3 @k2  L8 q8 r6 @a3 @h,17 @s,6 r*1
(t2)    r6 @1 o3 v15 p3 @k-3 L8 q8 r6 @a3 @h,17 @s,6 r*1
(t3)    r6 @1 o4 v15 p1 @k-5 L8 q8 r6 @a3 @h,17 @s,6
(t4)    r6 @1 o4 v15 p2 @k9  L8 q8 r6 @a3 @h,17 @s,6
(t5)    r6 @4 o4 v12 p1 @k-2 L8 q8 r6 @m6 @h30 @s5
(t6)    r6 @4 o4 v12 p3 @k-2 L8 q8 r6 @m6 @h30 @s5
(t7)    r6 @4 o5 v12 p2 @k-2 L8 q8 r6 @m6 @h30 @s5
(t8)    r6 @2 o0 v16 p3 @k0  L8 q7 r6 @o8
(t9)    r6    o0 v9  p3      L8 q7 r6 @r1
(t10)   r6    o2 v9  p3      L8 q4 r6 @r1
(t11)   r6 @1  o2 @u127  v15   p3          L8  q6 r6 @r1
(t12)   r6 @19 o4 @u127 @v107  p3    @k0  L8  q8 r6
(t13)   r6 @55 o4 @u107 @v93  @p66   @k0  L8  q8 r6
(t14)   r6 @55 o4 @u109  v0    p3    @k4  L8  q8 r6
(t15)   r6 @35 o4 @u76  @v70  @p64  @k-3 L8  q8 r6
(t16)   r6 @80 o3 @u98  @v96  @p111 @k-4 L8  q8 r6
(t17)   r6 @80 o3 @u98  @v96  @p17  @k6  L8  q8 r6
(t18)   r6 @11 o5 @u111  v8    p3   @k-6 L8  q8 r6
(t19)   r6 @29 o4 @u108 @v69   p3   @k0  L8 @q1 r6
(t20)   r3 L24
(t21)   r6 @1  o4 @u80  @v54  @p55  @k5  L8  q8 r6
(t22)   r6 @2  o1 @u114 @v111 @p100 @k0  L8  q8 r6
(t23)   r6 @2  o0 @u127 @v111 @p28  @k80 L8  q8 r6

/--------------------------------

/ ギターカード(SN-U110-07)がない場合の設定

/(t16)  r6 @58 o3 @u108 @v96  @p111 @k-4 L8  q8 r6
/(t17)  r6 @58 o3 @u108 @v96  @p17  @k6  L8  q8 r6

/--------------------------------

(t16)   @c7 @a96,+2,+2,+2,+3,+5,+7,+9=0 r*2
(t17)   @c7 @a96,+2,+2,+2,+3,+5,+7,+9=0

/--------------------------------
/       Rhythm

/       Noise (Bass Drum)
/       OPM
(t8)    [do]|:4c.c.r4.c4r4|c4:|r4
(t8)    |:4c.c.r4.c4r4c4:|
(t8)    |:|:16c.c.|r4.:|rc4
(t8)    |:3c.c.r4.:|c.c.|r4.:|rc4
(t8)    |:|:c.c.r4.c4.rc4:|||:4c.c.|r4.:|rc4:|
(t8)    |:4c.|c.r4.:|r*60c4
(t8)    |:|:8c.c.|r4.:||rc4:|r4.
(t8)    |:c.c.r4.c4.rc4:||:4c.c.|r4.:|rc4
(t8)    c.c.r4.c4.rc4c.r*204c4
(t8)    r*60c16r*60c.c4r.c.c*60c.r.c16c4.rc4

/ トラック9,10はどちらか一方だけ入れて
/ コピーと修正をすれば楽です。

/       ADPCM Bass
(t9)    [do]|:4c.c.r4.c4.|rc4:|{rrr}8.r.
(t9)    |:4c.c.r4.c4.|rc4:|{rrcrrr}4.
(t9)    |:
(t9)    |:8c.|c.r4.:|c*17{rr}*19rc4
(t9)    |:8c.c.|r4.:|rc16r.
(t9)    |:3c.c.r4.:|c.|c.r{rrrrr}4:|
(t9)    c*17{rr}*19rc4
(t9)    |:|:c.c.r4.c4.rc4:|||:4c.|c.r4.:|
(t9)    c*17{rr}*19rc4:|
(t9)    |:4c.|c.r4.:|r.{rrcrrr}4.
(t9)    |:8c.|c.r4.:|c*17{rr}*19rc4
(t9)    |:8c.c.|r4.:|rr16r.
(t9)    |:c.c.r4.c4.rc4:||:4c.|c.r4.:|
(t9)    c*17{rr}*19rc4
(t9)    c.c.r4.c4.rc4c.r*168r*34c*14r.
(t9)    r*60c16r*60c.c4r.c.c*60c16r*60c16c4.
(t9)    {rrcrrr}4.

/       ADPCM Snare
(t10)   [do]|:4r.r.v9d4.r4.|dr4:|{v5dv6dv7d}8.v9d.
(t10)   |:4r.r.d4.r4.|dr4:|{v6dv5dv6ddv7dd}4.v9
(t10)   |:b*0
(t10)   |:8r.|r.d4.:|r*17{v4dv6d}*19v9dr4b*0
(t10)   |:8r.r.|d4.:|dr*10v8d*25v9b*13
(t10)   |:3r.r.d4.:|r.|r.d{v6dv3dv4dv6dv7d}4v9:|
(t10)   r*17{v4dv8d}*19v9dr*35b*13
(t10)   |:|:r.r.d4.r4.dr4:|||:4r.|r.d4.:|
(t10)   r*17{v4dv8d}*19v9dr*35b*13:|
(t10)   |:4r.|r.d4.:|r.{v6dv5dv6ddv7dd}4.v9b*0
(t10)   |:8r.|r.d4.:|r*17{v4dv6d}*19v9dr4b*0
(t10)   |:8r.r.|d4.:|dv8d*10v9d*25b*13
(t10)   |:r.r.d4.r4.dr4:||:4r.|r.d4.:|
(t10)   r*17{v4dv6d}*19v9dr*35b*13
(t10)   r.r.d4.r4.dr4r.r*168d.r*10d*26b16
(t10)   r*60r16d*60r.r4d.b.r*60r16d*60r16r4.
(t10)   {v6dv5dv6ddv7dd}4.v0d*0

/       LA Hihat & Cymbal
(t11)   [do]|:64f.:|
(t11)   |:a.|:29f.:|fg4a.|:30f.:|fa+16
(t11)   r.|:13f.:||f.f.:|fg.a16
(t11)   |:30f.:|fg.a16|:32f.:|
(t11)   |:'a.f'|:29f.:|f|g4:|r.a16
(t11)   |:30f.:|fg.a16
(t11)   |:15f.:|fa+16
(t11)   r.|:6g.:|'g.a'|:4g.:||:4f.:|

/       LA E.Tom (Left)
(t22)   [do]
(t22)   |:|:4b-f.f.c-4|{b-b-}f.f.c-4:|{b-b-ffd}*60d4..:|
(t22)   |:20r*132g-.e16c*108:|
(t22)   |:|:b-f.f.c-4{b-b-}f.f.c-4:|||:r*132g-.e16c*108:|:|
(t22)   r*132g-.e16c*108r2.{rb-b-ff}*60d4..
(t22)   |:8r*132g-.e16c*108:|
(t22)   |:b-f.f.c-4{b-b-}f.f.c-4:||:r*132g-.e16c*108:|
(t22)   b-f.f.c-4{b-b-}f.f.c-4
(t22)   r*132{b-b-}16b-f16r*276b-16f.r*216{b-b-b-fff}4.r4.

/       LA E.Tom (Right)
(t23)   [do]|:|:8r.|b+*60a-16a-a16:|r*108:|
(t23)   |:|:8r*132b.b16|a*108:|
(t23)   {b-b-}a-*84|:r*132b.b16a*108:|:|
(t23)   |:|:4r.b+*60a-16a-a16:|||:r*132b.b16a*108:|:|
(t23)   |:r*132b.|b16a*108:|r*120
(t23)   |:8r*132b.b16|a*108:|{b-b-}a-*84
(t23)   |:4r.b+*60a-16a-a16:||:r*132b.b16a*108:|
(t23)   |:r.b+*60a-16a-a16:|
(t23)   r2...b+.r*264b+a.r*324

/--------------------------------
/       Slap Bass

/       PCM
(t12)   [do]o1|:64f.:|
(t12)   |:|:8q8f.@q10f4rq8ff.<<q5_19f16~19>>q8f.fe-4:|
(t12)   @k-128
(t12)   |:2q8f.@q10f4rq8ff.<<q5_19f16~19>>q8f.f@k0e-4:|:|
(t12)   |:|:7d-.:||d-d-16&|:7d-.:|d-e-16
(t12)   |:7f.:|e-f16&|:6f.:|fe-.d-16&:|
(t12)   d-.>|:8b-.:|<@q3|:c.c.r.c.r.c.r4.:|
(t12)   |:8q8f.@q10f4rq8ff.<<q5_19f16~19>>q8f.fe-4:|
(t12)   |:b-.a.a-.g.|f.g.a-.a.:|fg16a-b-.a-.a-16
(t12)   |:q8f.@q10f4rq8ff.<<q5_19f16~19>>q8f.fe-4:|
(t12)   b-.a.a-.g.f.g.a-.a.b-.r*216e-
(t12)   d-*72<d-*108>d-4d-.g-2.<g-2...

/--------------------------------
/       Sax

/ トラック1～4は、先頭と最後を除いてまったく同じです。
/ トラック13だけが少し違います。

/       PCM
(t13)   [do]@55@v93r*2304@p66
(t13)   |:|:|:@q23a-.f.b-.f.q8a-@q23a-.b-.r.q8f16@q23
(t13)   |a-.f.a-.b-.q8<c*60>@q23b-.r.q8f16:|
(t13)   @q23<e-.d.c.>b-.q8a-a16a-f16r4.:||r*576:|
(t13)   ~4r*504q8|:f@q23a-.<@q4f*72>q8b*138r*18@q23b-.a-.|
(t13)   <@q2f*60>q8b*108r16@q23b-.
(t13)   q8a-b-16r*20<c*52>fa-16r*432:|
(t13)   q6<a-*60>q8b*132{rb}b-@q23a-.<@q23c.r.
(t13)   q8c16&@b-683c@b0@q23c.e-.c.q8c4r.
(t13)   r*60c16g>b16<fe.c16>bb-*60r
(t13)   @v93r*2256|:f16q6a-q8b-2.a-16b-<c16r.|
(t13)   r>f16@q2a-q8b-16|:3@q2bq8b-16:|@q2a-q8f16
(t13)   @q2b-q8a-16r*60@q23e-.q8f*156r*384:|<r2q8d-16@q2
(t13)   dq8f16@q2dq8d-16@q2c>q8b-16@q2a-q8f16<@q2c>q8b-16
(t13)   r.<e-q8f*542@43r*10@v65@q1p1o5'b-<f''b-16<f'

/       OPM
(t1)    [do]@v120r*2304
(t1)    |:|:|:@q23a-.f.b-.f.q7a-@q23a-.b-.r.q8f16@q23
(t1)    |a-.f.a-.b-.q8<c*60>@q23b-.r.q8f16:|
(t1)    @q23<e-.d.c.>b-.q8a-a16a-f16r4.:||r*576:|
(t1)    ~1r*504q8|:f@q23a-.<@q4f*72>q8b*138r*18@q23b-.a-.|
(t1)    <@q2f*60>q8b*108r16@q23b-.
(t1)    q8a-b-16r*20<c*52>fa-16r*432:|
(t1)    q6<a-*60>q8b*132{rb}b-@q23a-.<@q23c.r.
(t1)    q8c16&c-@q23c.e-.c.q8c4r.
(t1)    r*60c16g>b16<fe.c16>bb-16^4r
(t1)    _1r*2256|:f16q6a-q8b-2.a-16b-<c16r.|
(t1)    r>f16@q2a-q8b-16|:3@q2bq8b-16:|@q2a-q8f16
(t1)    @q2b-q8a-16r*60@q23e-.q8f*156r*384:|<r2q8d-16@q2
(t1)    dq8f16@q2dq8d-16@q2c>q8b-16@q2a-q8f16<@q2c>q8b-16
(t1)    r.<e-@q10f*552@q1ff16

(t2)    [do]v15r*2304
(t2)    |:|:|:@q23a-.f.b-.f.q7a-@q23a-.b-.r.q8f16@q23
(t2)    |a-.f.a-.b-.q8<c*60>@q23b-.r.q8f16:|
(t2)    @q23<e-.d.c.>b-.q8a-a16a-f16r4.:||r*576:|
(t2)    ~1r*504q8|:f@q23a-.<@q4f*72>q8b*138r*18@q23b-.a-.|
(t2)    <@q2f*60>q8b*108r16@q23b-.
(t2)    q8a-b-16r*20<c*52>fa-16r*432:|
(t2)    q6<a-*60>q8b*132{rb}b-@q23a-.<@q23c.r.
(t2)    q8c16&c-@q23c.e-.c.q8c4r.
(t2)    r*60c16g>b16<fe.c16>bb-16^4r
(t2)    _1r*2256|:f16q6a-q8b-2.a-16b-<c16r.|
(t2)    r>f16@q2a-q8b-16|:3@q2bq8b-16:|@q2a-q8f16
(t2)    @q2b-q8a-16r*60@q23e-.q8f*156r*384:|<r2q8d-16@q2
(t2)    dq8f16@q2dq8d-16@q2c>q8b-16@q2a-q8f16<@q2c>q8b-16
(t2)    r.<e-@q10f*552<@q1ff16

(t3)    [do]v15r*2304@1p1
(t3)    |:|:|:@q23a-.f.b-.f.q7a-@q23a-.b-.r.q8f16@q23
(t3)    |a-.f.a-.b-.q8<c*60>@q23b-.r.q8f16:|
(t3)    @q23<e-.d.c.>b-.q8a-a16a-f16r4.:||r*576:|
(t3)    ~1r*504q8|:f@q23a-.<@q4f*72>q8b*138r*18@q23b-.a-.|
(t3)    <@q2f*60>q8b*108r16@q23b-.
(t3)    q8a-b-16r*20<c*52>fa-16r*432:|
(t3)    q6<a-*60>q8b*132{rb}b-@q23a-.<@q23c.r.
(t3)    q8c16&c-@q23c.e-.c.q8c4r.
```

```
(t3)    r*60c16g>b16<fe.c16>bb-16^4r
(t3)    _1r*2256|:f16q6a-q8b-2.a-16b-<c16r.|
(t3)    r>f16@q2a-q8b-16|:3@q2bq8b-16:|@q2a-q8f16
(t3)    @q2b-q8a-16r*60@q23e-.q8f*156r*384:|<r2q8d-16@q2
(t3)    dq8f16@q2dq8d-16@q2c>q8b-16@q2a-q8f16<@q2c>q8b-16
(t3)    r.<e-@q10f*552<@q1@4p1v14ff16

(t4)    [do]v15r*2304@1p2
(t4)    |:|:|:@q23a-.f.b-.f.q7a-@q23a-.b-.r.q8f16@q23
(t4)    |a-.f.a-.b-.q8<c*60>@q23b-.r.q8f16:|
(t4)    @q23<e-.d.c.>b-.q8a-a16a-f16r4.:||r*576:|
(t4)    ¯1r*504q8|:f@q23a-.<@q4f*72>q8b*138r*18@q23b-.a-.|
(t4)    <@q2f*60>q8b*108r16@q23b-.
(t4)    q8a-b-16r*20<c*52>fa-16r*432:|
(t4)    q6<a-*60>q8b*132{rb}b-@q23a-.<@q23c.r.
(t4)    q8c16&c-@q23c.e-.c.q8c4r.
(t4)    r*60c16g>b16<fe.c16>bb-16^4r
(t4)    _1r*2256|:f16q6a-q8b-2.a-16b-<c16r.|
(t4)    r>f16@q2a-q8b-16|:3@q2bq8b-16:|@q2a-q8f16
(t4)    @q2b-q8a-16r*60@q23e-.q8f*156r*384:|<r2q8d-16@q2
(t4)    dq8f16@q2dq8d-16@q2c>q8b-16@q2a-q8f16<@q2c>q8b-16
(t4)    r.<e-@q10f*552<@q1@4p2v14ff16

/--------------------------------
/       Sax & Organ

/ トラック14の6～11行目と
/ トラック18の2～7行目は同じです。

/       PCM
(t14)   [do]@55r*2304@u114@p27
(t14)   |:v0|:|:@q23e-.c.f.c.q8e-@q23e-.f.r.q8c16@q23
(t14)   |e-.c.e-.f.q8a-*60@q23f.r.q8c16:|
(t14)   @q23<c.>b-.a-.g.q8ff16fc16r4.@v93:||r*576:|
(t14)   @38@u106v9o4r*14p2r*504q8
(t14)   |:f@q23a-.<@q4f*72>q8b*138r*18@q23b-.a-.|
(t14)   <@q2f*60>q8b*108r16@q23b-.
(t14)   q8a-b-16r*20<c*52>fa-16r*432:|
(t14)   q7<a-*60>q8b*132{rb}b-q7a-.
(t14)   <q3c4q8rc16&c-@q23c.e-.c.q8c4r2
(t14)   c16g>b16<fe.c16>bb-*60
(t14)   r*4000@43@v65@q2r*582p2o6'cf''c16f'

/       LA
(t18)   [do]r*8012o5p1q8
(t18)   |:f@q23a-.<@q4f*72>q8b*138r*18@q23b-.a-.|
(t18)   <@q2f*60>q8b*108r16@q23b-.
(t18)   q8a-b-16r*20<c*52>fa-16r*432:|
(t18)   q7<a-*60>q8b*132{rb}b-q7a-.
(t18)   <q3c4q8rc16&c-@q23c.e-.c.q8c4r2
(t18)   c16g>b16<fe.c16>bb-*60
(t18)   r*4612

/--------------------------------
/       Stringth & Key.

/ トラック15と21は一部似たところがあります。

/       PCM
(t15)   [do]r*1152@v71
(t15)   'e-a-<c'288'dgb'288'd-g-b-'288'cfa'288
(t15)   q8|:r*2268'a-8<cf''b-<e-g'156<'d-4.fa-'
(t15)   'e-gb-'*60'd-*8fa-'v0r*28q6@v71'>a-8<cf'r*240:|q8
(t15)   _2>'fa-b'*288'fa-b'*288'f2.a-<c'
(t15)   'g2.b-<d''a-2.<ce-''g2.b-<d'
(t15)   'fa-b'*288'fa-b'*288@q2|:'e.a-<c'
(t15)   'e.a-<c'r.'e.a-<c'|r2.:|r.'e4.a-<c'
(t15)   r*4056
(t15)   @v66o4'a-<d-f'264@q8'b-<d-f'288q8¯4'b-<cf''b-16<cf'

/       LA
(t21)   [do]q8r*1152@v54o4
(t21)   'e-a-<c'288'dgb'288'd-g-b-'288'cfa'288
(t21)   q8|:r*2268'a-8<cf''b-<e-g'156<'d-4.fa-'
(t21)   'e-gb-'*60'd-*8fa-'v0r*28q6@v54'>a-8<cf'r*240:|q8
(t21)   >'fa-b'*288'fa-b'*288'f2.a-<c'
(t21)   'g2.b-<d''a-2.<ce-''g2.b-<d'
(t21)   'fa-b'*288'fa-b'*288@q2|:'e.a-<c'
(t21)   'e.a-<c'r.'e.a-<c'|r2.:|r.'e4.a-<c'
(t21)   r.q8@3o3
(t21)   |:4|:r*60p1q6@v90'e-f'rp3q8@v116'>b->b-''c16>c'
(t21)   q5'e-.>e-'p1@q1@v92'f.b-<d'q6'e-a-<c'p3q8
(t21)   |r16'e-.fa-':|'e-4fb-':|
(t21)   r*576
(t21)   |:r*60p1q6@v90'e-f'rp3q8@v116'>b->b-''c16>c'
(t21)   q5'e-.>e-'p1@q1@v92'f.b-<d'q6'e-a-<c'p3q8
(t21)   |r16'e-.fa-':|'e-4fb-'
(t21)   r*564
(t21)   @1@v55o3'a-<d-fa-<d-'264@q8'b-<d-fb-<d-'288
(t21)   @q2¯32<<'fb-<cf''f16b-<cf'

/       OPM
(t5)    [do]p1r*1152@v116
(t5)    e-*288d*288d-*288c*288
(t5)    |:@1o4@m0
(t5)    @v0|:|:@q23e-.c.f.c.q8e-@q23e-.f.r.q8c16@q23
(t5)    |e-.c.e-.f.q8a-*60@q23f.r.q8c16:|
(t5)    @q23<c.>b-.a-.g.q8ff16fc16|r4.@v121:|r8.
(t5)    @4o4q8p1@v116@m6a-8b-*156<d-4.e-*60
(t5)    d-16_14d-8>¯14a-8
(t5)    @z102,-4,-4,-4,-4,-4,-4,-4a-*240@z:|
(t5)    @v116
(t5)    @3p1_4|:f*288f*288|f2.g2.a-2.g2.:|
(t5)    @4p1¯1|:e.e.r.e.|r2.:|r.e4.
(t5)    r*4056¯5o4a-*264@q8b-*288q8r.

(t6)    [do]r*1152@v114
(t6)    a-*288g*288g-*288f*288
(t6)    q8¯2|:r*2268<c8e-*156f4.g*60
(t6)    f16_14f8¯14c8
(t6)    @z100,-4,-4,-4,-4,-4,-4,-4c*240@z>@v114:|
(t6)    @3_3|:a-*288a-*288|a-2.b-2.<c2.>b-2.:|
(t6)    @4_1|:a-.a-.r.a-.|r2.:|r.a-4.
(t6)    r*4056¯5o5d-*264@q8d-*288q8r.>

(t7)    [do]p2r*1152@v116
(t7)    c*288>b*288b-*288a*288<
(t7)    q8¯2|:r*2268f8g*156a-4.b-*60
(t7)    a-16_14a-8¯14f8
(t7)    @z102,-4,-4,-4,-4,-4,-4,-4f*240@z@v116:|
(t7)    @3p2_3|:>b*288b*288<|c2.d2.e-2.d2.:|
(t7)    @4p2_1|:c.c.r.c.|r2.:|r.c4.
(t7)    r*4056¯5o5f*264@q8f*288q8r.

/--------------------------------
/       Overdrive Guit.

/       PCM

/ トラック16と17では (A),(B) それぞれ似ていますが、
/ ベンド(@b)の値が違うので注意。

/(A)
(t16)   [do]|:|:|:>q8f.<q3f.q8re-16f|a-4q3f.rq8f16e-f16:|
(t16)   |>@b-43,640,33b-.&@b640,-43,0b-16
(t16)   @b-1408b-@b-43b-.a-4<:|b-.b16a-b-.a-4:|
/(B)
(t16)   _5q8|:|:|:|:>f.f.@d1b-*18@b1322r*18
(t16)   @b3370r*60@d0b-16@b-43<|cr16e-f4:|
(t16)   |q4fq8r16b-a-4:||q4cq8r16>b-a-4<:|cr16>b-a-.
(t16)   g4g.@d1b-*18@b1322r*18@b3370r*60
(t16)   @d0b-16@b-43<q4cq8>r16b-a-4f.f.r*216<:|
(t16)   ¯5r*2076c16g>b16<fe.c16>b=2(b-,e-)84,40=0@v96
(t16)   |:|:|:f.f.@d1b-*18@b1322r*18@b3370r*60@d0b-16@b-43
(t16)   |<cr16e-f4>:|q4|<fq8r16b-a-4>:|<cq8r16>b-a-4:|
(t16)   @58_10r*1692o4e-8@q10f*552r@80r16¯10

/(A)
(t17)   [do]|:|:>q8f.<q3f.q8re-16f|a-4q3f.rq8f16e-f16:|
(t17)   |>@b64,747,33b-.&@b747,64,0b-16
(t17)   @b-1301b-@b64b-.a-4<:|b-.b16a-b-.a-4
(t17)   |:|:>q8f.<q3f.q8r>b-16<c|e-4q3c.rq8c16>b-<c16:|
(t17)   |>@b64,747,33b-.&@b747,64,0b-16&
(t17)   @b-1301b-@b64b-.a-4<:|f.f+16e-f.e-4
/(B)
(t17)   _5q8|:|:|:|:>f.f.@d1b-*18@b1429r*18
(t17)   @b3477r*60@d0b-16@b64<|cr16e-f4:|
(t17)   |q4fq8r16b-a-4:||q4cq8r16>b-a-4<:|cr16>b-a-.
(t17)   g4g.@d1b-*18@b1429r*18@b3477r*60
(t17)   @d0b-16@b64<q4cq8>r16b-a-4f.f.r*216<:|
(t17)   ¯5r*2076c16g>b16<fe.c16>b=2(b-,e-)84,40=0@v96
(t17)   |:|:|:f.f.@d1b-*18@b1429r*18@b3477r*60@d0b-16@b64
(t17)   |<cr16e-f4>:|q4|<fq8r16b-a-4>:|<cq8r16>b-a-4:|
(t17)   @58_10r*1692o4e-8@q10f*552r@80r16¯10

/ ギターカードがない時はトラック16，17の
/ 最終行だけを以下のように変更して下さい。

/(t16)  _15r*1692o4e-8@q10f*552rr16¯15
/(t17)  _15r*1692o4e-8@q10f*552rr16¯15

/--------------------------------
/       LA Seq & Key.

(t19)   [do]|:32'f<f''f16<f''<f16<f''f16<f''<f16<f':|
(t19)   r*8064
(t19)   @3o1p2|:4
(t19)   |:@v103q8f.f.r_10<q4fr4fr4|cr:|q8e-4:|
(t19)   @q1@29o4@v69
(t19)   |:8'f<f''f16<f''<f16<f''f16<f''<f16<f':|
(t19)   @3o1p2
(t19)   |:@v103q8f.f.r_10<q4fr4fr4|cr:|q8e-4
(t19)   @q1@29o4@v69
(t19)   |:4'f<f''f16<f''<f16<f''f16<f''<f16<f':|
(t19)   r*864

/       Panning
/ (A)
(t20)   [do]
(t20)|:5@p 60r@p 56r@p 52r@p 48r@p 44r@p 40r@p 36r@p 32r
(t20)   @p 28r@p 24r@p 20r@p 16r@p 12r@p  8r@p  4r@p  0r
(t20)   @p  4r@p  8r@p 12r@p 16r@p 20r@p 24r@p 28r@p 32r
(t20)   @p 36r@p 40r@p 44r@p 48r@p 52r@p 56r@p 60r@p 64r
(t20)|  @p 68r@p 72r@p 76r@p 80r@p 84r@p 88r@p 92r@p 96r
(t20)   @p100r@p104r@p108r@p112r@p116r@p120r@p124r@p127r
(t20)   @p124r@p120r@p116r@p112r@p108r@p104r@p100r@p 96r
(t20)   @p 92r@p 88r@p 84r@p 80r@p 76r@p 72r@p 68r@p 64r:|
(t20)   r*10368
/ (A)
(t20)|: @p 60r@p 56r@p 52r@p 48r@p 44r@p 40r@p 36r@p 32r
(t20)|  @p 28r@p 24r@p 20r@p 16r@p 12r@p  8r@p  4r@p  0r
(t20)   @p  4r@p  8r@p 12r@p 16r@p 20r@p 24r@p 28r@p 32r
(t20)   @p 36r@p 40r@p 44r@p 48r@p 52r@p 56r@p 60r@p 64r
(t20)   @p 68r@p 72r@p 76r@p 80r@p 84r@p 88r@p 92r@p 96r
(t20)   @p100r@p104r@p108r@p112r@p116r@p120r@p124r@p127r
(t20)   @p124r@p120r@p116r@p112r@p108r@p104r@p100r@p 96r
(t20)   @p 92r@p 88r@p 84r@p 80r@p 76r@p 72r@p 68r@p 64r:|
(t20)   r*576
/ (A)'
(t20)   @p 60r@p 56r@p 52r@p 48r@p 44r@p 40r@p 36r@p 32r
(t20)   @p 28r@p 24r@p 20r@p 16r@p 12r@p  8r@p  4r@p  0r
(t20)   @p  4r@p  8r@p 12r@p 16r@p 20r@p 24r@p 28r@p 32r
(t20)   @p 36r@p 40r@p 44r@p 48r@p 52r@p 56r@p 60r@p 64r
(t20)   @p 68r@p 72r@p 76r@p 80r
(t20)   r*864

/--------------------------------

(t1)    [loop]
(t2)    [loop]
(t3)    [loop]
(t4)    [loop]
(t5)    [loop]
(t6)    [loop]
```

```
(t7)    [loop]
(t8)    [loop]
(t9)    [loop]
(t10)   [loop]
(t11)   [loop]
(t12)   [loop]
(t13)   [loop]
(t14)   [loop]
(t15)   [loop]
(t16)   [loop]
```

```
(t17)   [loop]
(t18)   [loop]
(t19)   [loop]
(t20)   [loop]
(t21)   [loop]
(t22)   [loop]
(t23)   [loop]

(p)
```

リスト13

```
-ROLLING THUNDER2- TERROR FOR SALE (C)namco  by ENG 92/06/28 (+PCM8,CM64&07)

-- counter --
 1:00000041 00003A80   2:00000041 00003A80   3:00000040 00003A80   4:00000040 00003A80
 5:00000040 00003A80   6:00000040 00003A80   7:00000040 00003A80   8:00000040 00003A80
 9:00000040 00003A80  10:00000040 00003A80  11:00000040 00003A80  12:00000040 00003A80
13:00000040 00003A80  14:00000040 00003A80  15:00000040 00003A80  16:00000042 00003A80
17:00000040 00003A80  18:00000040 00003A80  19:00000040 00003A80  20:00000040 00003A80
21:00000040 00003A80  22:00000040 00003A80  23:00000040 00003A80
```

リスト14

```
/ -ROLLING THUNDER2- TERROR FOR SALE (C)namco
/   Programmed by ENG 92/06/28 (+PCM8,CM64&07)

1 = gate_sd.pcm,v40
2 = rvbs1.pcm,v170,m1
3 = 2,v40
4 = gatek.pcm,v75
```

```
.o0c = teck.pcm,v65,m4
.o2d = 2,m3,d700
.o2b = crsh0.pcm,v42

.erase 1
.erase 2
.erase 3
.erase 4
```

Z-MUSICへの移植

さて，X68000のZ-MUSICで書かれた曲を他機種に持っていくのはちょっとたいへんすぎますが，他機種用のMIDIデータをZ-MUSICで演奏するのはさほど困難ではありません。

ここではX1 版のMIDIシーケンサで記述されたセーラームーンの曲をZMSファイルに変換する際の手順について解説しましょう。この曲はX1の内蔵音源(というかPSG)を使っていないので完全にコンバートすることができます。ZMSファイルでなくX-BASICへ直してもいいのですが，FORループは|: :|のループに置き換えればプログラムの必要はありませんので作業の楽なZMSファイルに変更します。

まず，初期設定，そしてトラックバッファの確保，チャンネルのアサインといったお約束のものを並べます。

とりあえずPLAY文はそのまま打ち込んでください。ここから変換作業に入ります。

まず，チャンネルを分離します。PLAY文中での“:”はチャンネルごとのセパレータを意味します。エディタで“:”を“:^J”に置換していったんセーブしてください。再ロードすると改行に変わっているはずです。こうして分離されたトラックをPLAY文のところから，

(T1)
(T2)
(T3)
⋮

という具合に先頭にトラック指定を加えていきます。基本的にPLAY文ごとに(T1)からこの指定をつけていきますが，PLAY文の最後が“;”で終わっているところは，その直後のPLAY文は続きの番号からにしなければいけません。

次に命令の置き換えです。

I	→	@	音色切り替え
M	→	N	チャンネル切り替え
<	→	>	オクターブダウン
>	→	<	オクターブアップ
V	→	@U	ベロシティ設定
P	→	@P	パンポット指定
^	→	Y	コントロールチェンジ
¥	→	@B	ベンド指定
-	→	@U＋	相対ベロシティ
__	→	@U−	相対ベロシティ

に置換します。さらに音色切り替えIの数値から100を引いてください。パンポット値は，

@P＝(P－8)×16

に，ベンド値は，

@B＝(¥－128)×64

に変換してください。これでデータの基本的な変換は終わりです。そのほかのものは同じ仕様です。

あとは同一PLAY文内の各トラックの音長を手作業で揃えていきます(休符を加える)。基本的にZ-MUSICのほうが高速なので，音色切り替えなどの（楽器にとって）重い処理を同時に行っているところでは楽器がデータを取りこぼしてしまいます。休符を調整してできるだけタイミングをずらせてください。

なお，曲の最初にプログラムでコントロールチェンジなどを各チャンネルに送っていますが，これは各トラックの先頭に“@V100”を加えれば省略可能です。

変換例（参考）

```
==================  S_moon4.zms  ==================
.comment Sailor Moon アイキャッチ for CM-64

(i)

(b1)
(m1,2000)(a1,1)
(m2,2000)(a2,2)
(m3,2000)(a3,3)
(m4,2000)(a4,4)
(m5,2000)(a5,5)
(m6,2000)(a6,6)
(m7,2000)(a7,7)
(m8,2000)(a8,8)
(m9,2000)(a9,9)

(t1)@v100T140n2@89y64,64O3L8rB<_4ET145_A<_DT150_G<_C2.&
(t2)@v100n11@36r1r2
(t3)@v100n11@36r1r2
(t4)r1@v100r2
(t5)r1@v100r2
(t6)r1@v100r2
(t7)r1@v100r2
(t8)r1@v100r2
(t9)r1@v100r2

(t1)T162R16&R4y64,0
(t2)O6L8@b-3520C16&@b-2816C16&@b-2112C16&@b-1408C16&@b-704C16&@b0Q7C2.>EF<
(t3)r1r4r16
(t4)n10@01O2R16D4L4CDCD
(t5)n10@01O2R16G+4L8<C+>F+G+F+F+F+G+F+
(t6)n12@18O1R16R4L8AAAAAAAA
(t7)n3@74@u30O4R16R4C1&
(t8)n4@74@u30O3R16R4F+1&
(t9)n13@11@u80O4L8R16R4EEE.EEE16EE

(t1)r1
(t2)>Q7F+2.<Q6C4
(t3)r1
(t4)CDCD
(t5)F+F+G+F+F+F+G+F+
(t6)A-A-A-A-A-A-A-A-
(t7)C1
(t8)F+1
(t9)E-E-E-.E-E-E-16E-E-

(t1)r1
(t2)>Q6A4Q4B8Q8B8&B1<
(t3)O5Q6F+4Q4G8Q8G8&G1
(t4)DC8D
(t5)F+G+F+<C+>
(t6)Q7F+4Q5G8Q8G8&G1
(t7)>Q6A4Q4B8Q8B8&B1<
(t8)>Q6F+4Q4G8Q8G8&G1<
(t9)F+4G8G8&G1

(p)
```

コンピュータアーキテクチャ編

加算器回路の工作実習

前回までの理論をふまえて，いよいよ製作実習に取りかかります。電子工作が初めての人でも挑戦できるように，工具を揃えるところから懇切丁寧に解説されています。じっくり本文を読み進めながら製作してみてください。

Misawa Kazuhiko
三沢 和彦

今月はいよいよ加算器の工作実習に進みたいと思います。今回製作する加算器自体は，TTL IC 1個の非常に簡単な回路なので工作自体は，すぐ終えられるものです。しかし，今回はアーキテクチャ編に入って初めての実習編なので，工具の揃え方からハンダ付けの方法まで，ごく基本的なところからゆっくり解説していきます。

工具の揃え方

電子工作をするには，まず工具を揃えなければなりません。工具はあり合わせのものを使うよりも，きちんと最初から揃えることを勧めます。最低限必要な工具は，ハンダゴテ，ニッパ，ラジオペンチ，カッター，ワイヤストリッパ，ピンセット，ドライバの7点です。

まずはこの7つ道具について，順番に説明していきたいと思います。

1）　ハンダゴテ

電子回路の配線には，ハンダゴテがいちばんよく使われます。最近ではIC工作用にラッピングワイヤという方法もかなり普及してきましたが，IC以外の部品の配線に使うとなると，まだまだハンダゴテのほうが便利です。ハンダゴテは消費電力別で分類されており，太い配線をするに従って電力の大きいものが必要になってきます。ただし，ICの足の間隔は0.1インチ（＝2.54mm）しかないので，コテ先が太いものでは隣の足もいっしょにハンダ付けしてしまうことになります。IC工作などの細かい配線をするには，コテ先が細くて20W程度のものがベストです。私は太洋電気産業（グット）のCX-20を愛用しています。

表1　電子工作に必要な工具（参考）

ハンダゴテ			
	グット	CX-20	1,850円
ニッパ			
	グット	YN-16	2,000円
	HOZAN	N-32	1,400円
ラジオペンチ			
	グット	YP-10	1,450円
	HOZAN	P-36	1,100円
ワイヤストリッパ			
	グット	YS-2	1,450円

また，ハンダもそれに合わせて細いものを選ぶのですが，必ず「ヤニ入り」のものを選んでください。このヤニは，ハンダを溶かしたときにハンダの中からハンダ付けしようとする導線部分に流れ出てきて，その表面をきれいにし，ハンダのノリがよくなるようにしてくれます。

コテ先はいつもきれいにしておかないと，上手にハンダ付けすることができません。そこで，スポンジ付きのコテ台も用意しておきましょう。スポンジには水を含ませ，そこでコテ先に付いた余計なハンダを拭き取ります。私はコテと同じメーカーのST-30というタイプを使っています。

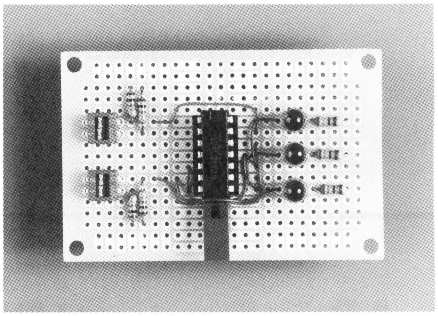
余裕のある人は，ハンダ吸い取り器も用意しておきたいものです。これは，ハンダ付けに失敗したとき，余計にこびりついたハンダを吸い取ってくれるものです。これさえあれば，多少の失敗も恐れずに安心して工作することができます。私はグットのGS-20を使っています。

2）　ニッパ

3）　ラジオペンチ

4）　ワイヤストリッパ

ニッパとラジオペンチとワイヤストリッパとは，兼用することもできますが，やはりそれぞれ役割があるので別々に買い揃えたほうがよいでしょう。専用の工具を使うと作業の能率が格段に違います。



繰り上がり付き加算器回路

ニッパは配線材を切断するときに頻繁に使います。IC工作用には，先が細く全体として小さめのニッパが適当です。ハンダ付けした部品の足を切り取るときなどは，ニッパの先が細くないと，うまく切ることができません。

ラジオペンチは，導線を折り曲げたり金属ケースなどを加工したりするときに使います。また，少し太めの針金を切ったりするときも，ニッパよりラジオペンチを使います。これは，IC工作に使うような比較的精密なニッパでは，少しでも太めの導線を切ろうとすると刃先がすぐ傷んでしまうからです。

ワイヤストリッパは，被覆の付いた配線材の被覆を剥いたりするときに使います。これは，ニッパの刃先に異なる大きさの丸い溝を刻んだようなもので，そこに被覆線を当てて挟むと周りの被覆線だけ切り込みが入り，中の導線を切らずにすむのです。ニッパを使って被覆線の皮剥きをしようとすると，たいていの場合は中の導線まで傷をつけてしまいます。ハンダ付けをしてもそこから切れてしまったりして，配線ミスのもとになりやすいのです。そこで，ワイヤストリッパは，ぜひケチらずによいものを揃えてください。私はグットのYS-2とい

うものを使っています。

5） カッター

カッターは，文房具店で手に入るごく普通のものでかまいません。どちらかというと，刃の幅が太いもののほうがよいでしょう。

6） ピンセット

IC工作は細かい部品を扱うことが多く，また，ハンダゴテを当てると熱くて部品を直接持てないので，部品をつまみ上げるピンセットは必需品です。好みにもよりますが，先が尖ったもののほうが細かい部品を扱うのに便利なようです。これも特殊なものである必要はなく，文房具店で手に入るものでかまわないでしょう。

7） ドライバ

ドライバは，プラスドライバとマイナスドライバをそれぞれ大，中，小３本ずつぐらい用意しておくのがよいでしょう。

工具は以上のとおりに揃えれば，まず問題ありません。参考までに表１に主な工具のメーカー，型番と値段のリストを挙げておきます。

さて，工具は電子回路を組み立てるときに必要ですが，完成した回路の動作をチェックするためになくてはならないものがもうひとつあります。それは，テスターというものです。テスターは，１台で電流計，電圧計，抵抗計の役割をする測定機です。この連載においても，回路チェックをするときにどうしてもテスターを使う必要が出てくることもあります。もちろん実体配線図どおり間違いなく作れば，そのまま動作するはずです。やはりここはひとつテスターを揃えておくのが，これからハードウェアを志そうという人にとって，必要なことだと思います。

この連載でテスターを使うとすれば，TTL ICの各端子の電圧を計ってH/Lをチェックし，設計どおりの論理演算がなされているかを追っていく場合がほとんどです。また，きちんとハンダ付けされているかの導通チェックや，逆に余計なところがショートしていないかのチェックなど，基本的な回路チェックに欠かせません。

テスターは値段も性能もさまざまなものが市販されており，初めて選ぶ人には迷ってしまうところもあるかもしれません。基本的なのはアナログ式（指針式）で，電流，電圧，抵抗だけを測定できるものから，デジタル読み取りでコンデンサの容量やトランジスタの性能測定までできる，高級なものまであります。もちろん高級なものを買うに越したことはありませんが，初心者であれば欲張らずに，まずはごく普通のアナログ式で最低限の電流，電圧，抵抗が測定できるものでかまわないでしょう。私はSANWAというメーカーのSP-10Dという，もっとも基本的なタイプのもので十分重宝しています。

ハンダ付けの方法

ハンダ付けが下手だと，きちんと配線しているつもりでも回路がうまく動作してくれないことが多くあります。以下に，基本的なハンダ付けの手順を説明しますので，少し練習してみてください（図１）。

1） ハンダゴテのコテ先を水の含んだスポンジでよく拭いてきれいにする

きれいなコテ先でなければ，必ず失敗します。

2） コテ先にハンダを薄くメッキする

3） ハンダ付けしたいところにコテ先を当てて，十分温度を上げる

最大のポイントはこの部分です。ハンダを先にハンダゴテに付けてしまったら，絶対にうまくいきません。まず，ハンダ付けしたい部分の温度を上げ，ハンダをコテ先で溶かすのではなく，ハンダ付けしたい部分の温度で溶かすようにするのです。

4） 温度が上がったと思うところで，ハンダ付けしたい部分にハンダを当てる

ここでは，ハンダ付けしたい部分とコテ先の接触している部分にハンダを当てるのがベストです。

5） ハンダが流れたらすぐにハンダゴテを離さず，ハンダの表面張力でハンダ付けしたい部分がきれいに覆われるまで待つ

6） コテ先を離し，十分温度が下がるまで動かさずに待つ

7） 特に基板に配線材をハンダ付けするときには，付けたい導線の先にあらかじめハンダをのせておく

一度熱の通ったハンダは二度と使えないので，常にハンダ付けする瞬間に新しいハンダが，ハンダ付けしたいところに流れていくように心掛けます。いったん，ハンダゴテの先に溶けてのってしまったハンダでさえも，もう使いものになりません。スポンジでよく拭き取りましょう。

うまくできたハンダは，表面に光沢があります。失敗すると表面に艶がなくなり，ざらざらしていたり一部分だけ盛り上がっていたりして，いかにも取れてしまいそうに見えます。何度か納得のいくまで練習してみてください。

5V電源

TTL ICを使うを使うには，専用の＋5V電源が必要になります。この連載を続けていくうえで，5V電源を用意しなければなりませんが，とりあえずいくつかの方法を考えてみます。

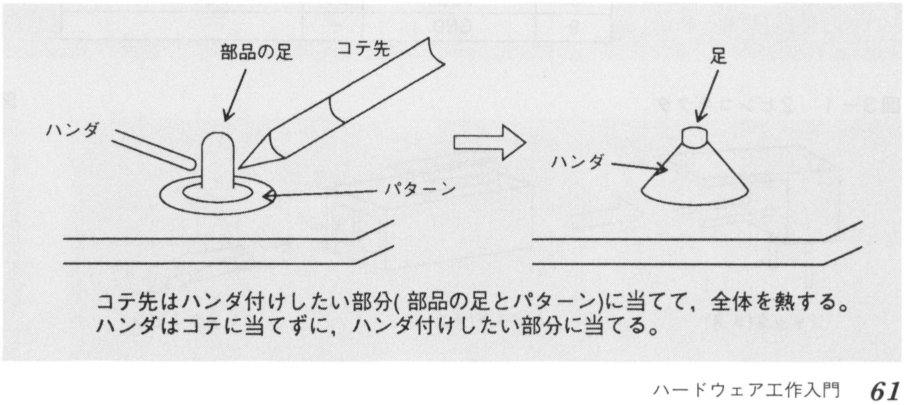
1） X68000のジョイスティックポートから取る

2） 専用電源を自作する

3） 専用電源を購入する

このうちもっとも簡単なのはもちろん3)です。市販のものを買ってしまえば，その日から問題なく使えます。しかしながら，市販の電源は高価なものが多く，またこの連載で扱うようなTTL ICの簡単な回路を駆動させるための電源としては，高級すぎ

図１　ハンダ付けのコツ



るものばかりです。そこで，いちばんのお勧めは1)ということになります。これには，専用ケーブルを１本工作するだけでOKです。必要な部品は表２の部品表にあるとおりです。

以前のハードウェア工作入門を読んでいただいた読者には，汎用の10ピンフラットケーブルがあり，最初はそれを流用してもいいかとも思いました。しかし，最近になって読み始めた方が多いことでしょう。とりあえず，5V電源だけ取るという目的だったら，ケーブルの工作は非常に簡単なものですんでしまいますので，改めて作り直すことにします。

X68000のジョイスティックポートは，外側から見て図２のような配列になってます。５番ピンが+5V，９番ピンがGNDになっています。９ピンのＤサブメスコネクタの５番ピンと，９番ピンに少し太めのビニール線をハンダ付けし（+5VとGNDとでビニール線の色を変えたほうがよいでしょう），反対側に２ピンコネクタプラグを取り付けます。この電源端子には，日本航空電子（JAE）というメーカーの基板用コネクタを用います。このコネクタは非常に着脱が簡単で，しかも汎用性があるので，IC回路基板では非常によく使われるものです。

このコネクタは，プラスチック製のプラグとジャック，そのほかにアルミのコンタクトピンという部品からなっています。プラグ，ジャック，コンタクトピンは，図３－1のような構造になっていて，導線をコンタクトピンにハンダ付けするようになっています。コンタクトピンは10本ひと組みで売られており，１本ずつ切り離して使います。導線をハンダ付けしたコンタクトピンは，プラグにある穴に裏から差し込みます。差し込むときに上下があるので，注意してください。コンタクトピンにあるひげが，ジャックの溝に引っ掛かるように差し込むのです。

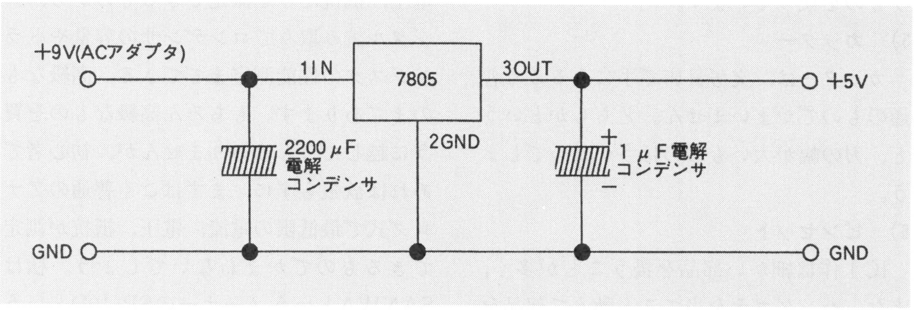
また，２ピンプラグはコンタクトピンを差し込むほうから見て右側が＋５V，左側がGNDなので，間違えないように差し込んでください。言葉ではわかりづらいかもしれませんが，図３－2をよく見て理解してください。

このように，ハンダ付け箇所は全部で４箇所という極めて簡単な工作で，専用電源ケーブルが出来上がりました。

さて，専用ケーブルを使うには，それだけのためにX68000の電源を入れて，かつジョイスティックポートを占有しなければなりません。これではあまりスマートな方法ではありませんね。今後の連載で工作していく回路を駆動するのに，すべて+5Vが必要になってくるので，いまのうちから専用電源を作ってしまうというのもテです。

また，X68000を所有していない読者もいるでしょうから，ここから作り始めるのもハードウェア工作の連載としては正しいあり方かもしれません。専用電源といってもメインになるのは，３端子レギュレータというIC１個です。実際の回路図は図４のとおりです。この回路の説明していくと長くなってしまいそうなので，来月改めて専用電源の製作を解説したいと思います。

図４　＋５Ｖ専用電源



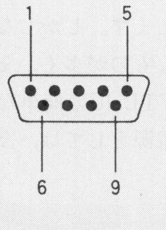
加算器回路と部品

今回工作する加算器の回路図を改めて書き下したものが図５です。部品数も少なく，非常に簡単に工作できる回路です。

まずは，個々の部品について説明してみましょう。２ビットDIPスイッチは，小さなスイッチが横に２列並んでいて，それぞれのスイッチに対応する端子が２本ずつ，計４本付いています。スイッチをONにするとその２端子間が導通し，OFFにすると断線します。今回の回路では，スイッチをショートすると入力は０（Ｌレベル）になるように対応していますので，基板に取り付けるときには，ON/OFFの向きと0/1の向きとを逆にしなければなりません。

TTL ICは，直接基板にハンダ付けする

図２　ジョイスティックコネクタピン配列



端子No	信号名	I/O	備　　考
1	IOA0	IN	8255のPA０端子
2	IOA1	〃	〃　PA1〃
3	IOA2	〃	〃　PA2〃
4	IOA3	〃	〃　PA3〃
5	Vcc1	OUT	+5V
6	IOA5	I/O	8255のPA5/PC6端子
7	IOA6	I/O	〃　PA6/PC7〃
8	IOC4	OUT	〃　PC4　〃
9	GND	−	

図３－１　２ピンコネクタ

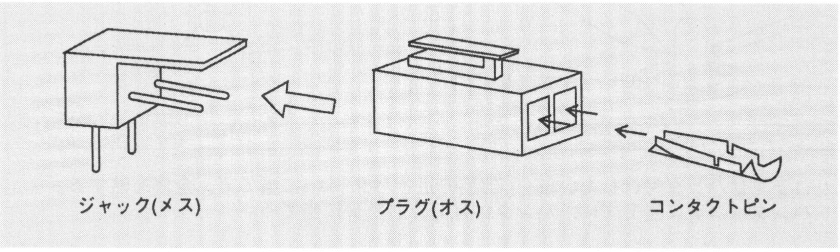


図３－２　コンタクトピンのハンダ付け

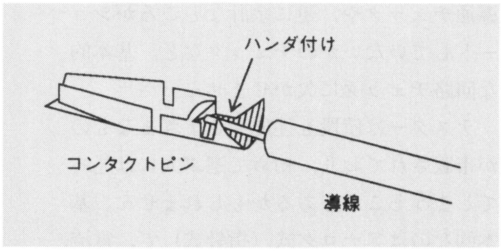
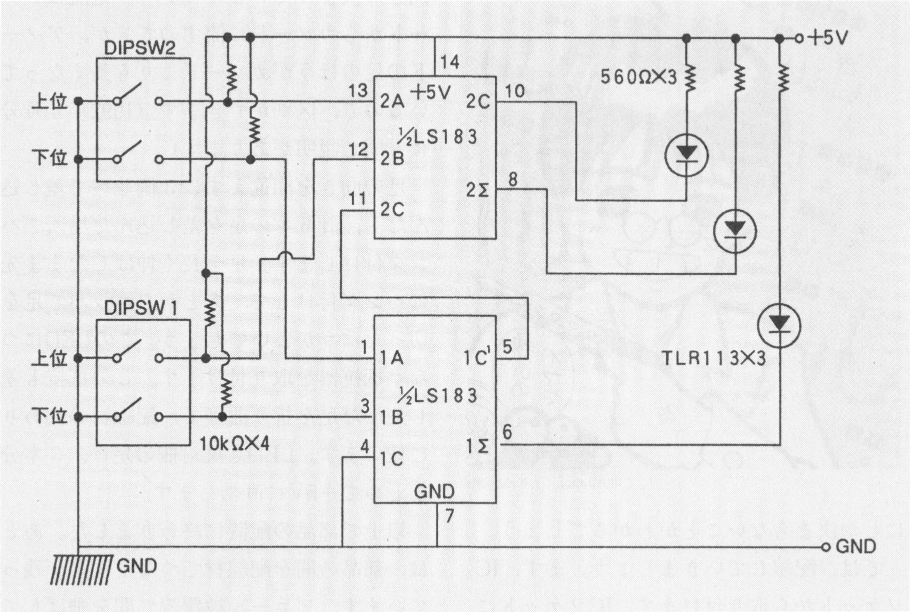


表2　部品表

専用電源ケーブル		
9ピンDサブメスコネクタ	1個	200円
IL2ピンコネクタオス（日本航空電子JAE）	1個	40円
コンタクトピン	1組	80円
加算器		
LS183	1個	230円
14ピンICソケット	1個	30円
2ビットDIPスイッチ	2個	@80円
IL2ピンコネクタメス（日本航空電子JAE）	1個	20円
TLR113A	3個	@20円
10kΩ抵抗	4本	
560Ω抵抗	3本	
スズメッキ線	少々	
ビニール配線材	少々	

図5　加算器回路図



よりもICソケットを使うことを勧めます。万一，間違って取り付けてしまった場合，一度ハンダ付けしたICを取り外すのは，非常に大変です。また，ハンダ付けに慣れないうちは，何度もハンダゴテを当てているうちにTTL ICを壊してしまうこともなきにしもあらずです。さらにソケットを使っておけば，同じICをほかの回路に使いまわすこともできます。以上のような理由から，基板にはICソケットをハンダ付けして，ICはあとから差し込むことにします。

LEDはTLR113Aを使いましたが，特にどの品種でなければならないというわけではありません。LEDには赤，黄，緑の3色があり，型番もそれぞれ，TLR(Red)，TLY(Yellow)，TLG(Green)となっています。このうち赤がもっとも明るく光って見えるので，よく使われます。ところで，今回の回路で注意すべきなのは，LEDが点灯しているときが数字の0に，消灯しているときが1に対応しているという点です。

電源入力端子は，今回工作した専用5V電源（あるいは専用電源ケーブル）に対応して，日本航空電子（JAE）というメーカーの基板用コネクタジャックを用いました。この連載で工作する回路は，電源入力端子としてすべてこのジャックを使うことにしています。

基板はサンハヤトのICB-87というIC専用基板で，この基板はこれまでの連載でも毎回のように使われてきた基板です。この基板では，TTL IC1個分のIC用パターンと周辺部品を取り付けるスペースとがあります。そして，中心に＋5VとGNDのラインが通っていて配線に便利なようになっています。ICを取り付ける部分には，3個ずつつながったパターンがプリントされていて，ICの足まわりの配線がくっついてしまわないように工夫されています。

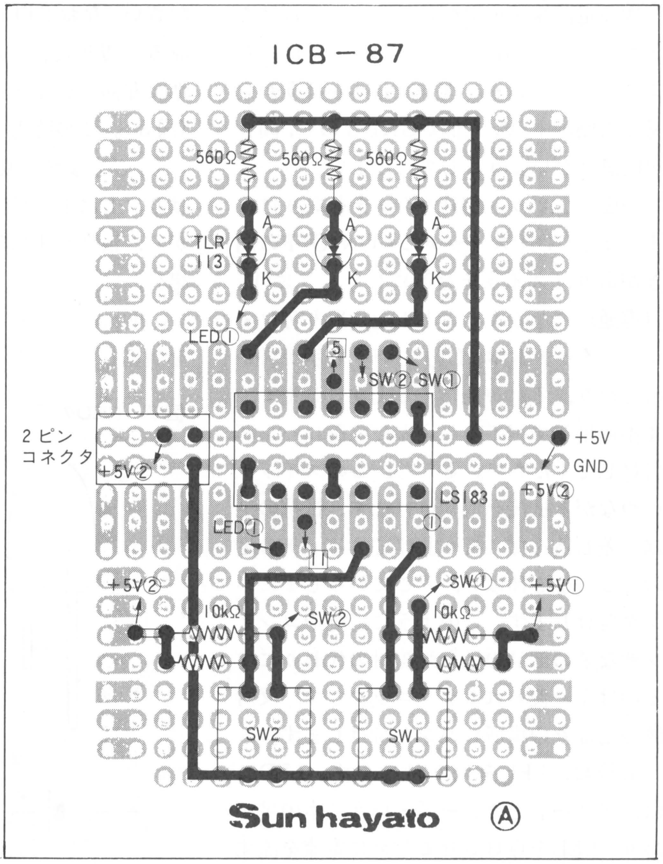
加算器回路の工作

では，実体配線図（図6）を参考にしながら，工作の手順を説明していきましょう。

まずは部品のレイアウトから決めていきます。今回はLS183 1個だけなので，ICの位置は選択の余地がほとんどありません。

また，電源用コネクタも＋5VとGNDとの向きを考えると場所は決まります。入力用の2ビットDIPスイッチは2個並べて基板の下のほうに配置します。DIPスイッチには10kΩ抵抗がセットになっていますので，2ビット×2個＝4本の抵抗を並べますが，スペースの関係で実体配線図のように両側向きで2本ずつ並べます。出力は繰り上がりを考慮すると3ビットになるため，各ビットに対応させたLEDを3個並べます。それぞれのLEDに560Ω抵抗が1本ずつセットになっていますので，LEDと抵抗は各ビットの組みごとに縦に並べます。以上のようにレイアウト決めを言葉で考えると複雑そうに見えますが，実体配線図を眺めてみると，直感的にほとんどひととおり

図6　実体配線図



illustlation：Y. Kawahara

にしか決まらないことがわかるでしょう。

では，配線していきましょう。まず，ICソケットから取り付けます。ICソケットにも向きがあるので，注意してください。7番ピンはGND，14番ピンは＋5Vなので，内側に折り曲げて，基板の真ん中を通っている2本ラインへ，それぞれいっしょにハンダ付けしてしまいます。LS183の4番ピンもGND直結なので，やはり内側に折り曲げてハンダ付けしてしまいます。残りのピンは，そのままそれぞれのパターンにハンダ付けします。

次に電源端子を取り付けます。これはちょうど＋5VとGNDラインにハンダ付けします。2ピンジャックは差し込むほうから見て右側が＋5V，左側がGNDなので，向きを間違えないようにしてください。

今度は2ビットDIPスイッチを取り付けましょう。DIPスイッチにもON/OFFの向きがあり，スイッチを下にしたときにON（導通）になるように上下逆に取り付けます。スイッチ本体にOFFという表示が書かれており，それが上にくるようにします。4本の足をすべて差し込んだ位置にハンダ付けしてください。そして，このスイッチにつながる抵抗を取り付けます。抵抗の足は，差し込んだら折り曲げてスイッチの足のところまで伸ばし，そこにいっしょにハンダ付けします。このように抵抗やコンデンサなどの部品の足を配線材の代わりにするのはよく使われるテクニックです。反対側の足はその場所にハンダ付けします。

最後に，LEDを取り付けます。LEDには，アノードとカソードという2本の足があり，LEDを点灯させるために電流を流す向きが決まっています（図7）。電流はアノードからカソードに流すのですが，アノードの足のほうがカソードよりも長くなっているので，区別ができます（1992年4月号にも同じ説明があります）。

足の向きを間違えずに3個並べて差し込んだら，計6本の足を差し込んだ場所でハンダ付けします。足を長く伸ばしたまま先にハンダ付けして，あとからニッパで足を切ったほうがよいでしょう。このLEDにつなぐ抵抗器を取り付けます。この抵抗も差し込んだ足を折り曲げて，配線材の代わりに使います。LEDと反対側の足は，3本分まとめて＋5Vに直結します。

以上で部品の配置は終わりました。あとは，部品の間を配線材でつなぐ作業が残っています。ビニール被覆線で間を飛ばしてつなぐものが8本，スズメッキ線で基板に這わせてつなぐものが4本あります。実体配線図上の対応をよく確認しながら，ハンダ付けしていってください。

動作チェック

LS183をソケットに差し込み，DIPスイッチを全部下にして電源をつないでみてください。なおTTLICの向きは，ICを上から見ると切り欠きがあり，その切り欠きを基準に左回りに1～14番ピンになっています（図8）。この時点でLEDがすべて点灯したら，ひとまず成功です。DIPスイッチの下は0，上は1に対応していて，LEDは点灯が0，消灯が1に対応しています。いまの場合，

00＋00＝000

を演算していることになります。

次に，DIPスイッチのどちらか一方の右側の桁を上に上げてみてください。LEDの右端だけ1個消えましたか？　そうなれば，大成功です。これは，

00＋01＝001

を演算していることになります。あとは，任意の加算が実現できますので，試してみてください。

ではうまく動作しなかったときのチェックポイントを挙げてみましょう。

1）　LEDがまったく点灯しない

これは電源系統をチェックしてみてください。＋5VとGNDとが逆になっているかもしれません。あるいは，LEDの足の向きが逆の可能性もあります。まさか，LS183自体が逆に差されているようなことはありませんか。チェックしてみてください。

2）　DIPスイッチを動かすと，LEDが切り替わるが正しく演算しない

LS183の入出力端子が，DIPスイッチまたはLEDの正しい桁に対応していないことが考えられます。これは単なる配線ミスなので，回路図と実体配線図をしっかり追っていって間違いを見つけてください。

＊　　＊　　＊

今月でやっと繰り上がり付き2桁加算器が完成しました。こんなに簡単な回路でも，皆さんが日頃使っているX68000のいちばん基本となる部分なのです。このような基本的な回路を無数に組み合わせて，パーソナルコンピュータというものが出来上がっているのです。

では，今後どのような回路を組み合わせていけば，X68000のような複雑なハードウェアになっていくのか？　これは来月以降のお楽しみです。

図7　LEDの足

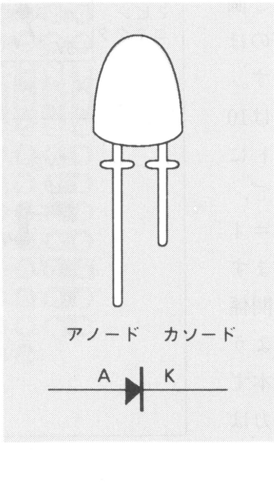
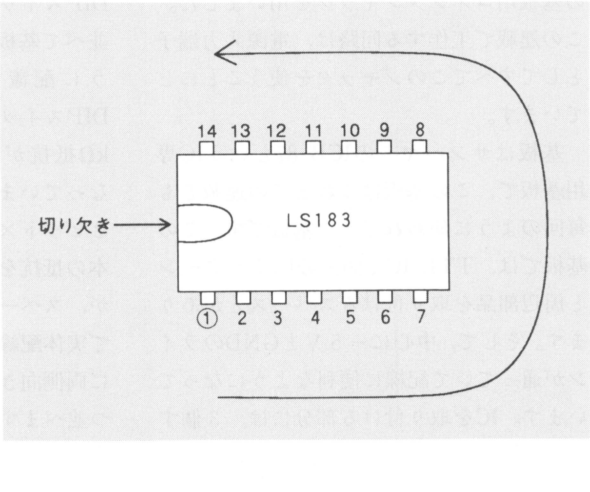


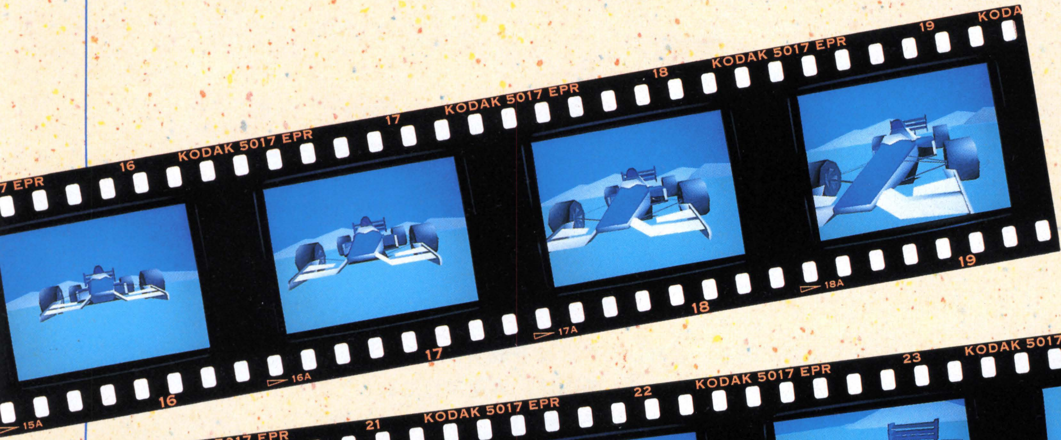
図8　TTL ICの足

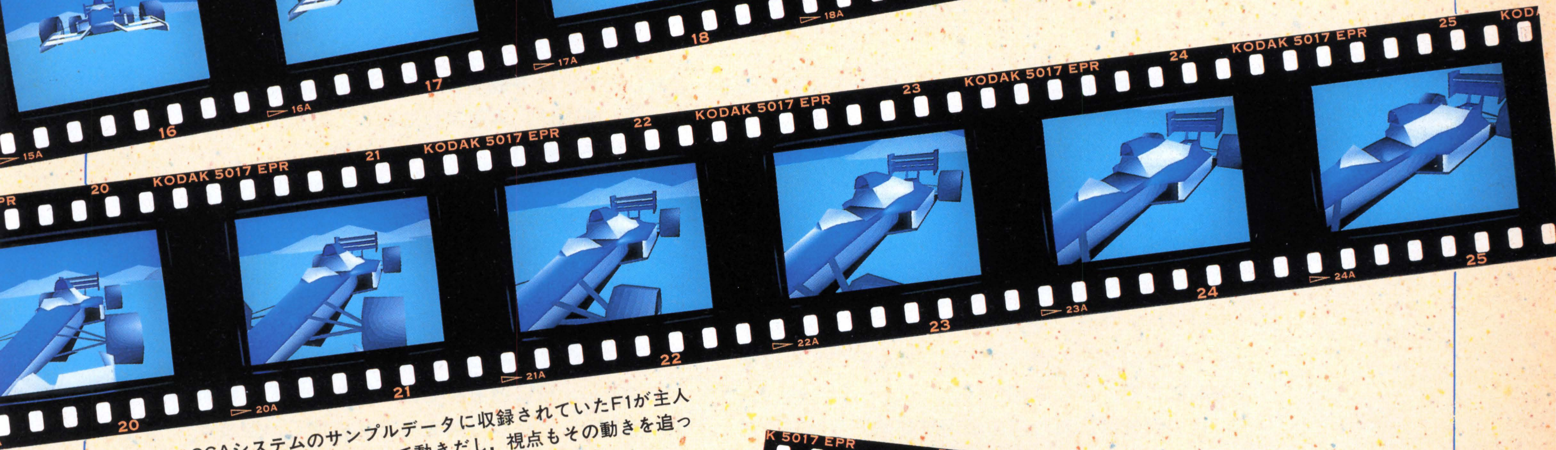


Oh!X Graphic Gallery

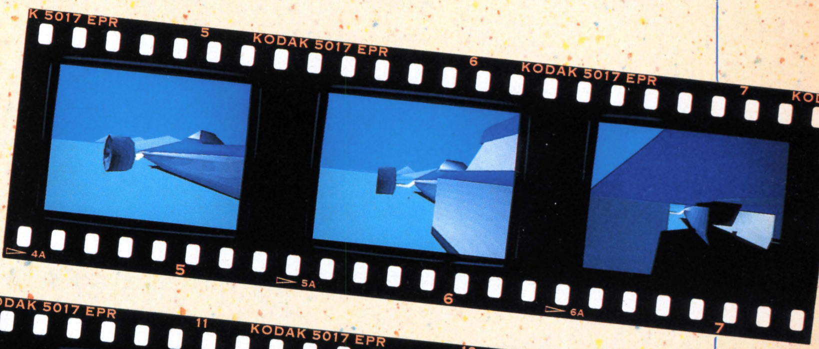
DōGA CGアニメーション講座

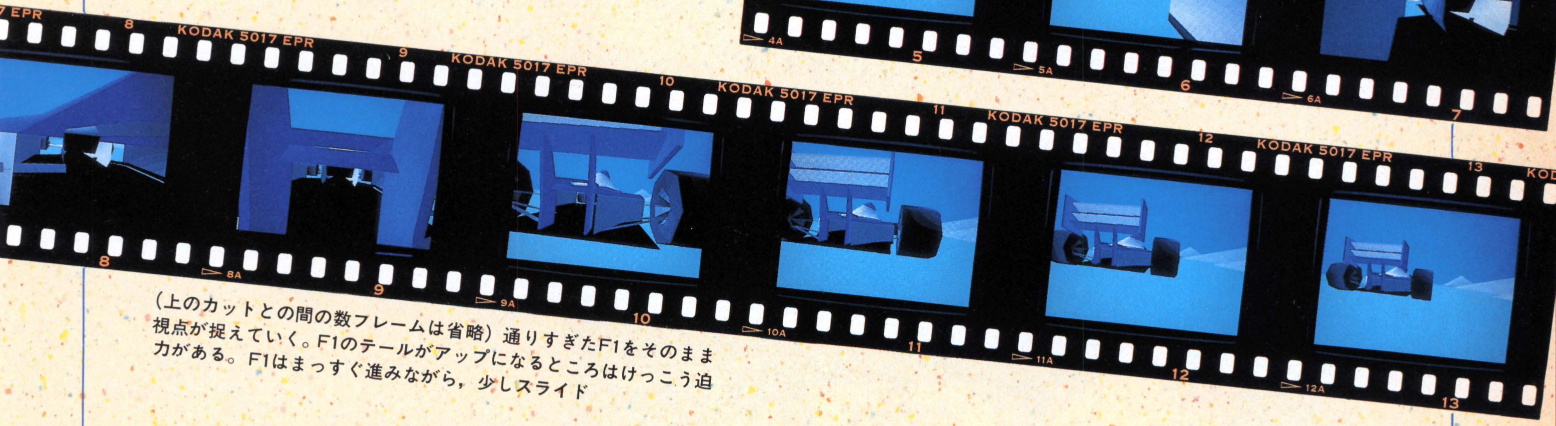
今月のDōGA CGアニメーション講座で作成するサンプルCGAを連続写真で紹介します。モデリングは行わないので，動きをしっかりと確認して，各自制作したモーションに間違いがないかどうかをもう一度確かめましょう。





DōGA CGAシステムのサンプルデータに収録されていたF1が主人公。まず，F1がこちらに向かって動きだし，視点もその動きを追っていく。視点の位置自体は変わっていない





（上のカットとの間の数フレームは省略）通りすぎたF1をそのまま視点が捉えていく。F1のテールがアップになるところはけっこう迫力がある。F1はまっすぐ進みながら，少しスライド

（同様に数フレーム省略）どんどん遠ざかっていくF1。THE END

[第17回]
キャラクタ

寺尾　響子

響子inCGわ～るど

自分だけのオリジナルキャラクタをつくってみたいと考えたことはありませんか?

“つくりたいけど，つくれない!”って?

……わかります，わかります，その気持ち。かつては私もそうでしたから。

オリジナルキャラクタをつくる

ロクハチがあります。ペイントソフトや3Dレンダリングソフトも買いました。真っ白の初期画面を前にして，手にはマウス。

この状況は，真っ白の画用紙を前にクレヨンを手にしているのと，なんらかわりません。さて，どうしよう。

ものを表現するのにあたって，これといったきまりはありません。好きなように表現すればよいのです。しかし，そのことで私たちは逆に迷ってしまいます。どこから手をつけたらいいのか，さっぱりわからないのです。

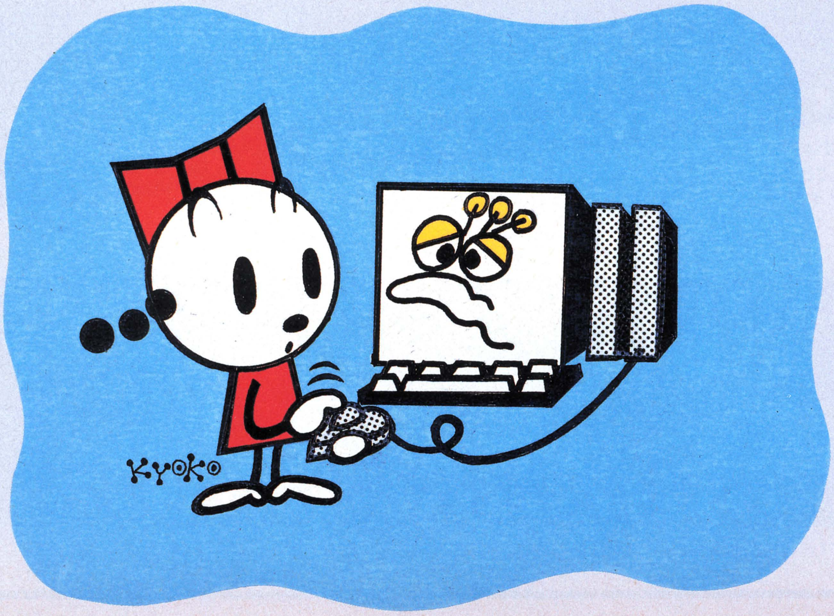
新しいキャラクタをつくる場合，私はまず，身近なひとや動物を観察したり，思い出したりしてみます。その際，外見よりもむしろ性格や行動を考えているときのほうが多いようです。

たとえば，いっしょにいるとウキウキした気分になれる，夏の空に向かって咲くひまわりのような背の高いひと。いつ出会っても塀の上で寝てばかりいて，撫でてもあくびしか返さない太った三毛猫。ドラえもんより気のきく，ステンレス製の小さなロボット，などのように……。

いったん，紙におおよその外見を写していきます。それから性格や行動を考えあわせて，強調するところと省略するところをピックアップして，デフォルメします。太った三毛猫はより太くし，ずぶとさを印象づけるために，つり目気味にしたりします。

こうした自分なりの見方をつけ加えることで，オリジナルのキャラクタができていきます。

キャラクタづくりについては，まだまだお話ししたいことがたくさんありますが，別の機会に譲



ることにしましょう。

＊　　　＊　　　＊

いままでに出会った，キャラクタを扱った作品で好きなものは数多くあります。ウォルト・ディズニーの作品はもちろんのこと，ハンナ・バーバラ，手塚治虫，宮崎駿，鳥山明などなど，きりがありません。そのなかで，最近こころに残っているものが2つあります。

ひとつはレイモンド・ブリッグスの「スノーマン」。絵本を原作としたアニメーションで，色鉛筆のタッチがやさしい作品です。

そしてもうひとつは，ユーリ・ノルシュテインの「話の話」[1] です。ちょっと変わったオオカミの子供が出てくる，モノトーンに近い淡い色彩のアニメーションで，見終わったあとに静かな空間が自分のまわりに広がります。

スノーマンは，CMでオンエアされたりキャラクタグッズが発売されているので，すでに知っているひとは多いと思います。ノルシュテインのほうは少し入手しづらいかもしれません。レンタルビデオでもあまり見かけません。

もし，みなさんの近くに持っているひとがいたら，ぜひ貸してもらって，ご覧いただくことをおすすめします。

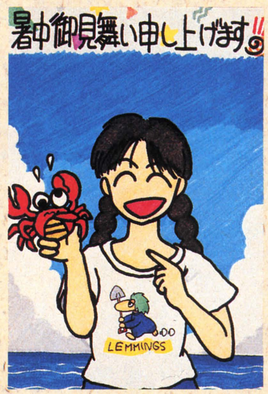
1）「話の話」，ユーリ・ノルシュテイン，発売元 レーザーディスク株式会社，価格 8,800円(税込)，型番 SF088-1026

暑中見舞いだ！
Oh!X reader'sぎゃらりぃ

暑い暑い，本当に暑かった季節は過ぎてしまいましたが，皆さん元気に過ごせましたか。数こそ少ないけど今回もきれいな暑中見舞いをありがとう。ハガキをくれた皆さん，これからもよろしくね。

▲見浦　崇(長野県)

▲橋本　和典(東京都)



▲石田　伯仁(神奈川県)

▲杉本　秀昭(宮城県)

▲中光　雄二(広島県)

▲上田　考一(福岡県)

FROM STAFF

Takahashi Tetushi

Kawahara Youi

▲溝畑　知幸(兵庫県)

▶玉野　健一(奈良県)

試用レポート

X68000用CD-ROMドライブを使う

Kioi Makoto 紀尾井 誠

ついに発売されたX68000対応のCD-ROMドライブ。もちろん専用ドライバつきでMacintoshのファイルをアクセスするユーティリティなども用意されています。肝心の使い勝手はどのようなものでしょうか？

吾輩はCD-ROMである

計測技研からX68000用CD-ROMドライブが発売されました。もちろん，Human68k上でCD-ROMドライブを扱うためのソフトウェアも付属しています。

CD-ROMメディアの大まかな仕様はISO9660で定められたフォーマットにほぼ統一されています。それに対応したドライバがあれば，現在さまざまな機種用に発売されているCD-ROMの多くがアクセス可能になるわけです。

CD-ROMの特徴は，大容量であること，書き込みができないこと，通常のCDと同様に音声を再生できることなどです。特に大容量の割に低価格なメディアが制作できることから，ソフトウェアの配布媒体として非常に期待されています。

製品内容を見てみましょう。ドライブは東芝製のXM-3301Aが使われています。CD-ROMドライブとして見るとアクセスタイムはかなり速いほうです。アクセス時間はハードディスクの10～20倍でも，それほど気になりません。初期のFM TOWNSを見て，CD-ROMはとてつもなく遅いものだという印象を持っている人もいるかもしれませんが，普通に使う分には困らない速度で動作します。

CD-ROMの転送速度は秒間100Kバイト程度です。グラフィックなどの極端に大きなデータでない限り，それほど待たされることはありません。まあ，これはこんなもんでしょう。

さらに，この製品はキャッシュメモリを64Kバイト内蔵しています。CD-ROMは回転速度が遅いのでブロック読み込みをしないで隣接データのアクセスなどを行うと極端に遅くなることが考えられますが，この製品では特に気になる遅さはありません。

ソフトはまだない？

さて，X68000にCD-ROMを接続するメリットはなんでしょうか？

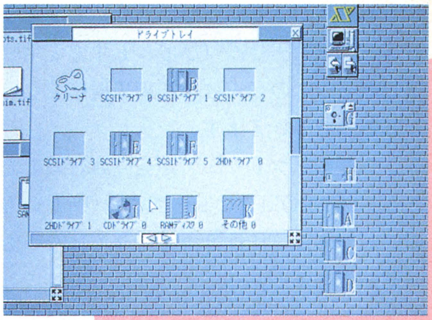
1) 他機種用に発売されているCD-ROMソフトが使用できる

2) 音楽用CDが再生できる

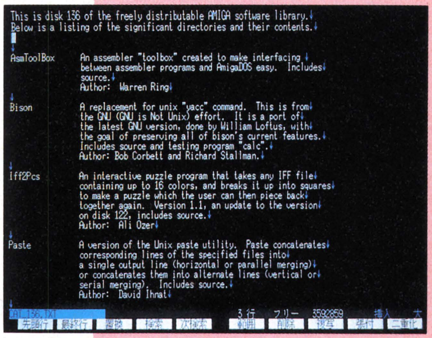
3) X68000用に発売されるCD-ROMソフトが使える

まず，1)ですが，ここで使用できるのは他機種用CD-ROMで使われているデータ部分です。プログラム部分については使用できません。データでも，標準MIDIファイルとか，TIFF画像とか，JPEG画像とか，テキストデータなどのフォーマットのはっきりわかったデータだけです。

CD-ROMはデータ供給媒体として多く使用されているので，これだけでも価値があるという方もいるでしょう。しかし，多くのソフトでは，独自のデータ構造を採用していたりしますので，それらについては各自でデータ構造を調べてアクセスプログ



ちゃんとCDのアイコンが使われている



普通のディスクと変わりなくアクセス可能

KGU-XCD(118,000円)

ラムを自作する必要があります。

2)はCDプレイヤーを持っていない人には朗報でしょうが，普通のCDプレイヤーを買ったほうがもっと安くすみます。

3)はとりあえず計測技研から2本発売される予定となっています。ひとつは，X68000用のフリーソフトウェアを集めた「フリーソフトウェアセレクション」，もうひとつは(CD-ROMソフトではありませんが)，「広辞苑検索ユーティリティ」です。

しかし，ドライブが出たからといって，CD-ROM用のソフトを開発するようなソフトハウスが（計測技研以外に）そうそうあるとは思えません（ひょっとすると電脳倶楽部？)。このへんは純正周辺機器でないことの弱みといえます。

いくつかソフトウェアが発売されるとしても，CD-ROM用に制作された「新しいもの」が現れる可能性は決して高くはないといえるでしょう。

とりあえず，ユーザーは1)の線で努力するしかありません。現在のところ，CD-ROMドライブに対する投資を回収することは難しいかもしれません。

使ってみる

では実際にX68000にCD-ROMドライブを接続して動作テストをしてみましょう。

最初に驚いたのはSCSIケーブルが同梱されていなかったことです。しかたないのでマシン室のハードディスクを1台殺して接続しました。

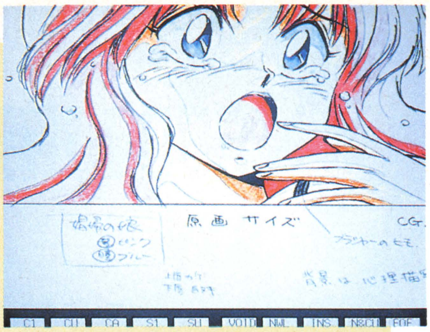
よく見るとコンセントがアースつきの三つ又のものでした。海外の製品ではありがちなことですが，ちゃんとしたお店なら変換アダプタをつけてくれるぞ，とぼやきつつ，薄目のタップに横差ししました。

ハードウェアのセッティングはこれで完了です。

次にソフトウェアのセッティングに移ります。CONFIG.SYSにCD-ROM用のデバイスドライバCDDEV.SYSを登録します。登録方法はSCSIドライバとほぼ同様です。リセットして起動……Human68k ver.2.03でないと怒られました。

手持ちの3.5インチFDDのドライバは2.03では動かないのですが，しかたないのでシステムを2.03に差し替えます。もう一度リセット。これでどうにかCD-ROMが扱えるようになりました。

CD-ROMの種類にもよりますが，通常のディスクドライブと同様にDIRやTYPEなどの命令が実行できます。使い勝手はハードディスクとほとんど変わりません。



各種CD-ROMから
各種CD-ROMから読み込んだ画像例。上はMacintoshの24ビットTIFF画像をTIFFLOAD.Xで表示（「SWIM SUIT CD」より)。右上はFM TOWNSの32768色TIFF画像をフリーソフトウェアのTIFF68K.Xで表示（「ALICEの館CD」より)。右はAMIGAのFredFish内にあったIFFファイル，HAMモード4096色画像のものをSV.Xで表示したもの。

どのようなCD-ROMが扱えるのか？

アンケートハガキなどを見るとX68000用のCD-ROMに過剰な期待を持っている人が多く見受けられます。残念ながら現在のところ他機種用のCD-ROMにアクセスできるというだけで，CD-ROMソフトを運用するところまでにはいたっていません。メディアにアクセスできることと，ソフトを利用できることには大きな隔たりがあります。たとえば現状で，PC-9801のディスクにアクセスできるからといって，PC-9801用のソフトを買ってきてファイルを読んだりデータを利用している人がはたしてどれくらいいるでしょうか？

それでもなお，CD-ROMで供給されるデータなどを利用することは十分考えられます。要は，ソフトウェアさえ作成すれば広辞苑検索ソフトのように，他機種用のCD-ROMでも有効利用が可能です。ここではそういった場合に使用できるCD-ROMを探してみましょう。

●FM TOWNSのCD-ROM

現在，もっとも入手しやすいCD-ROMといえばPCエンジンのROM²用またはFM TOWNS用のソフトウェアでしょう。ROM²はともかく，FM TOWNS用のCD-ROMは相性がよいのは当然でしょうか。日本語だから扱いも楽です。

CD-ROMとマルチメディア

「マルチメディアへの誘い」

近年，「CD-ROM＝マルチメディア」という誤った図式を広められたおかげで誤解している人がずいぶんいます。計測技研の広告もその例を出ていませんが，CD-ROMドライブの接続は単にファイルアクセスができるようになっただけの話でマルチメディアとは直接の関係はありません。実際，マルチメディアでは先端を走ってきたMacintoshやAMIGAでは本体のみですでにマルチメディアを実現しています。

では，マルチメディアとはなにかというと，広義には「文字情報以上のもの（グラフィックや音声）をいろいろ扱っています」ということを表すための言葉です。PC文化の土壌があってこその単語といえるでしょう。

CD-ROMは単なる記憶装置，マルチメディアはソフトウェア上の概念。実のところ，マルチメディア用のデータはでかくなりがち，という傾向以外に接点はないのです。マルチメディア対応のハードウェアにCD-ROMが搭載されているとはいえ，CD-ROMをつければマルチメディアが実現するというものでもありません。ごく低レベルな意味では実現するかもしれませんが，その程度のものはX68000登場時に実現されています。X68000というのはもとから結構マルチメディアマシンなのです。

今回のシステムではほぼそのまま読めます。理屈では非常にたちの悪いファイル名でもなければ完全にアクセスできるはずです。が，試してみたところ，一部のディレクトリにchdirできないものがありました。電脳倶楽部で発表されたCDFILE.Xではchdirできましたので，ファイル名上の問題ではないと思います。

FM TOWNS用のCD-ROMで，X68000に有用なものとしては，やはりGNUのCD-ROMでしょうか。たいていのものは読めますから，すべて利用可能ではあるのですが，非マルチメディア関係のCD-ROMのほうが利用価値は高いでしょう。

●MacintoshのCD-ROM

CD-ROMはファイルフォーマットがかなり統一されていることもあってか，Human68kがMS-DOSに近い構成のためか，CDDEV.SYSを使えばたいていのCD-ROMがそのまま読めてしまいます。しかし，うまくいかないのがMacintoshのCD-ROMです。MacintoshのCD-ROMをアクセスするためには専用ソフトが必要てす。

今回の製品にはSX-WINDOW用のSXMACDIR.Xというソフトウェアが付属しています。このソフトウェアはMacintoshのCD-ROMのディレクトリ構成を一覧したり，ファイルをHuman68kの形式で出力することができるものです。

このソフトにより，ほぼあらゆるMacintosh用データをX68000で読めるファイルに変換できます。CD-ROMの底にへばりついているQuickTimeとかいう400Kバイトのファイルを解析したい……という人には必携のツールといえるでしょう。

使い勝手はいまひとつといったところで，ファイル1個ずつ変換するだけでなくバッチ的にまとめて操作できればよかったのですが……。アクセスできないCD-ROMの場合ハングアップしてしまったり，いくつかバグも残っているようです。SX-WINDOW用しか用意されていないこと，ソースリストがないことが残念です。

速度的には，せっかく高速なドライブを使っているのにソフトが追いついていない感じです。1.8Mバイトくらいの画像ファイルをRAMディスク上に変換するのにだいたい70秒くらいかかります（16MHz時）。

また，このソフトウェアを使ってもすべてのMacintosh用CD-ROMがアクセスできるわけではありません。現在のところ，どのようなものがアクセス可能なのか詳しくわかっていませんので（ISO規格以前のものとも思えないのだが），各自で購入したCD-ROMを個別に試してみるしかありません。

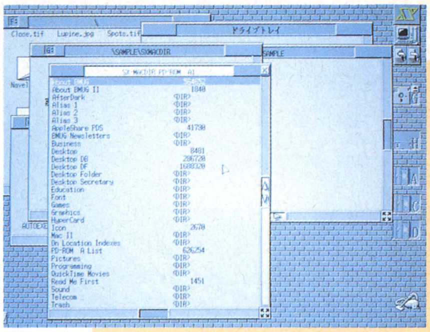
今回，こちらで試してアクセスできなかったものは「MATERIAL WORLD」というマッピング素材を集めたCD-ROMです。アクセスできたものは，いわゆる「PDS」を集めた「PD-ROM」，同様に映像を集めた「TV-ROM」などです。

いったんファイル出力するという方法を介してしかX68000でアクセスできないので，CD-ROMに付属するライセンス契約の内容を十分に確認する必要があります。海外製のものはたいてい現実的なセンで明記してあるのですが，データの一部複製も禁止するようなものの場合，このソフトウェアではアクセスできないことになります。

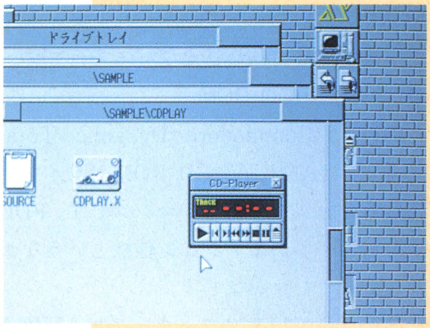
●PC用のCD-ROM

日本では入手しにくいものの，世界標準機用のCD-ROMというのもたくさん発売されています。

入手できるものをいくつか試したところ，すべてアクセス可能でした。ただし，今回試したものでは流用できそうなデータは見当たりませんでしたが。



Macintoshファイルビュア



CDプレイヤー

ついでに，MPC（MultiMedia PC）用のCD-ROMも使用してみました。これはマルチメディア対応のPCマシンでCD-ROMドライブを標準装備としています。これからのマルチメディアのひとつの核として注目されるメディアです。

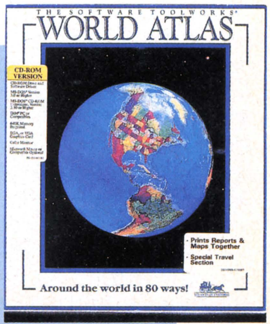
しかし，なぜか読めませんでした。MPCではなく普通のPC用のCD-ROMなら読めたので問題はないはずなのですが……。原因は不明です。

ちなみに今回試したMPC用のCD-ROMはマイクロソフトの「MultiMedia Beethoven Nineth Symphony」というもので，ベートーベンの交響曲9番に関する解説やクイズなどがまとめられたものですが，実に6トラック中の4トラックに「第九」がまるまる収録されていました。CDの容量がベートーベンの第九の長さで決められたというのは有名な話。そのため，このCD-

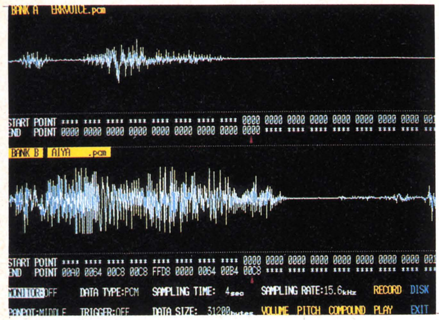
TV-ROM(Macintosh)

PD-ROM(Macintosh)

FredFish(CDTV)



WORLD ATLAS (IBM PC)



TOWNSのPCMデータを変換

ROMでは曲以外のプログラムやデータをすべて圧縮して詰め込んでいるそうです。

●CDTV用のCD-ROM

CDTV (Commodore Dynamic Total Vision) はAMIGAにCD-ROMドライブを搭載したマルチメディアマシンです。FDDやキーボードがオプション仕様というなかなか徹底したマシンです。

発売されているソフトウェアはAMIGAのゲームをCD-ROMにしたもののほか，大容量を生かしたビジュアルなデータ集などもたくさんあります。

多少不安はあったのですが，あっさりアクセスできました。アニメーションファイルなどもANIMファイルが多く，SV.Xで再生可能なものもありますので比較的利用価値は高いといえるでしょう。

入手は多少困難かもしれませんが，CDTV用の「FredFish Collection on CD-ROM」がおすすめです。これはAMIGAのFRS (Freely Redistributable Software) をまとめたもので，フロッピーディスク530枚分のFRSが収録されています（手元にあるバージョンでは480枚分)。一部のシステムディレクトリへのアクセスに難があるものの，カタログ，圧縮ファイルには問題なくアクセスできます。ファイルはLHarcで圧縮されているので，LH.XやLHA.Xを使えばX68000でも展開できるのです。

内容は多岐にわたっています。X68000で利用できそうなものはソースリストが付属しているプログラム，ANIM, MOVIE, IFFファイルなどのグラフィック，その他データ類に限られます。スプレッドシートのソースを見るとFORTRANだったりするワンダーなCD-ROMです。

アクセスの際の問題は，AMIGAとX68000ではファイル名の仕様が違うということでしょう。禁止された文字の使用，4文字以上の拡張子，長いファイル名，ピリオドで始まるファイル名……などが許されているためアクセスできないファイルが存在します。単に圧縮ファイルを展開する作業だけでも簡単にいかないこともあるでしょう。しかし，このようなファイル名が使用されているのはAMIGAのシステムファイルまわり(.infoなど)がほとんどなので，X68000ではアクセスしても意味がありません。そこで，たいていの場合は，

LHA E FFnnn ＊.???

のようにワイルドカードを使うことで，必要十分なファイルだけを取り出し可能です。

また，さらに入手しにくいかもしれませんが，ちゃんとMS-DOSでアクセスできるようなファイル名に変換された「FredFish Collection ON LINE CD-ROM」というバージョンも発売されているようです。

データを活用する

計測技研からドライブと一緒に送られてきたCD-ROMはMacintosh用の「SWIM SUIT CD」というやつでした。その名のとおり，水着のおねーちゃんのグラフィックが満載されたCD-ROMです（にーちゃんもいる)。ただし，TIFFビュアのようなものは送ってくれなかったので，丹氏に簡易TIFFローダを作ってもらいました。これはMacintosh形式の無圧縮TIFFファイルを画面内に適正な大きさで表示するものです。ただし，簡易版なので24ビットカラーのディザ化や「真面目な拡大/縮小」は行っていません。

一般的にいって，MacintoshのCD-ROMから持ってこれるのは画像データくらいのものでしょう。コンバータはデータフォークとリソースフォークを自動的に分離してくれるのですが，テキストデータの変換だけでも結構しんどい作業が必要になります。せめてテキストビュアとPICTビュアくらいは揃ってないと，MacintoshのCD-ROMを買ってきて云々どころではありません。

さらに，簡単に利用できそうなものとして，FM TOWNSのPCM音源で利用される＊.SNDファイルをX68000で使うためのコンバータを作成してみました。ただし，データの加工とAD PCM化にはZVT.Xが必要です。FM TOWNSのPCM音源は8ビットの無圧縮データなので，これをZVT.Xで扱う16ビットPCMデータに変換するだけのもので，なにも複雑なことはしていません。ヘッダ部分に書かれている数値の読み出しの際に数値がインテル並びになっているのを並べ直すことだけ気をつければよいでしょう。

なお，PCMバッファを64Kバイトしか持っていないFM TOWNSでも音声モード時には64Kバイト以上のデータが扱われることもあります。このプログラムではそういったデータには対応しておらず，最初の64Kバイト分だけを変換します。

将来に期待か？

これまでCD-ROMドライブが話題にのぼるたびに，書き込みができないことから光磁気ディスクの低価格化に期待するほうがよいのではないか……という議論が繰り返されてきたように思われます。

配布メディアとしてのCD-ROMは確かに大きな可能性を秘めています。確かに，X68000の性格からして，ユーザーがソフトウェア作成に取り組めたなら「凄いソフト」の登場も期待できたのですが……。残念ながら現状ではユーザーがCD-ROMソフト（CD-ROMのためのソフト）を作ることはかなり困難です(不可能ではありませんが)。

これまでシャープではCD-ROMに対して，接続は不可能ではない，という程度の対応しか示していませんでした。マルチメディア的使い方というのは最近になってようやく実現されてきたわけですから，単なる大容量メディアとしてのCD-ROMでは魅力は薄いと考えていたとしても不思議はないでしょう。

音楽用CDを演奏する

このCD-ROMドライブには音楽用のCDやCD-ROMに記録された音声トラックを再生するためにSX用のCDプレイヤーが付属しています。

これはごくオーソドックスなプレイヤーで，演奏開始/停止，スキップ，早送りなどの基本機能に，指定したトラックの演奏や連続演奏の機能を加えたのみで，ランダム選曲やプログラム演奏などはサポートされていません。これはソースリストもついてくるのでもっと高性能なプレイヤーを作成する際の参考にするとよいでしょう。なお，付属のプレイヤーではすべてのCDが演奏できるわけではないようです（曲数が多すぎるとダメ？)。

今後，おそらくシャープからCD-ROMに対するなんらかのアプローチがあるだろうことは予想できます。今後のCD-ROMの展開を予想すると，CD-ROM 1枚に1時間程度の動画像を収録することが常識化してくるのは間違いないでしょう。一部の新聞で開発中と報じられたAV指向の32ビットマシンでは，CD-ROMを扱うならDVIやMPEGチップセット（あるいはQuickTimeアクセラレータ）を使って動画像を扱うところまでやってきそうな気がします（あくまでも私見）。

いずれにせよ，こういったマルチメディア的な展開が行われてこそ，CD-ROMの価値は出てくるものと思われます。要はソフトウェアです。計測技研には現状を打開する魅力的なソフトウェアを開発していただきたいものです。

いろいろと厳しいことも書きましたが，こういったことをちゃんとわきまえて使う分には結構面白い周辺機器であることは確かです。もちろんユーザーがある程度増えれば展開もまた変わってきます。そのためには，ドライバソフトウェアの別売りなども必要かもしれません。今後の展開に期待しましょう。

リスト2

```
 10 /* 8ビットPCM→16ビットPCM変換
 20 str nam
 30 char dat(65535)
 40 int dat2(65535)
 50 int i,j,k,l,m,size,s
 60 /*
 70 input "fil name";nam
 80 i=fopen(nam+".snd","r")
 90 size=fseek i,0,2)-32
100 if size>65536 then size=65536
110 fseek(i,0,0)
120 for k=0 to 7
130     print chr$(fgetc(i));
140 next
150 print
160 print "ID           ="; ilp(4)
170 print "Data Size    ="; ilp(4)
180 print "Loop Point   ="; ilp(4)
190 print "Loop Length  ="; ilp(4)
200 print "Sampling Rate="; ilp(2)/98;"kHz"
210 print "音程修整=";       ilp(2)
220 print "音程=";fgetc(i)
230 fseek(i,31,0)
240 fread(dat,size,i)
250 fclose(i)
260 /*
270 j=fopen(nam+".pcm","c")
280 for k=0 to size-1
290     m=dat(k)
300     if m>127 then dat2(k)=65536-(m-128)*20 else dat2(k)=m*20
310 next
320 for k=0 to (size-1)/2
330     dat2(k)=((dat2(k*2))shl 16)+dat2(k*2+1)
340 next
350 fwrite(dat2,size/2,j)
360 fcloseall()
370 end
380 /*
390 func ilp(s)
400 int a,t=0
410 for a=1 to s
420     t=t+fgetc(i)*pow(256,a-1)
430 next
440 return(t)
450 endfunc
```

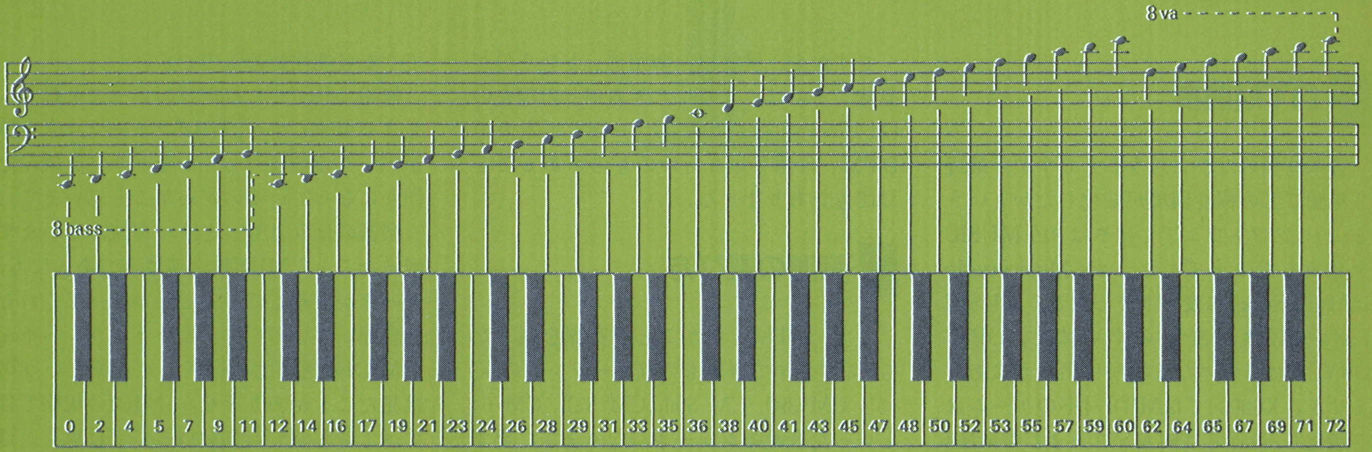
リスト1

```
/*
 *   tiffload.c
 *   - MacintoshのTIFFファイルローダ(簡易版)
 *      1992/08       丹 明彦
 */

#include   <stdio.h>
#include   <graph.h>
#include   <basic0.h>

#define   MAXWIDTH   1024            /* 予想される画像の最大幅 */

unsigned char  header[512];          /* これくらいあれば十分 */
unsigned char  rgbBuf[MAXWIDTH][3];  /* 1ラスタぶんのフルカラーデータ */
unsigned char  mBuf[MAXWIDTH];       /* 1ラスタぶんのモノクロデータ */
unsigned short rastBuf[512];         /* X68000のGRAMイメージ */

int resizeTable[2048];               /* これくらいあれば十分 */

short imageWidth = 0,                   /* 画像の横幅 */
      imageLength = 0,                  /* 画像の縦の長さ */
      bitPerPixel = -1,                 /* 1ピクセルのビット数 */
      compression = -1,                 /* 圧縮形式(非圧縮のみサポート) */
      photometricInterpretation = -1, /* 色モード(パレットあり/なし) */
      stripOffset = -1;                 /* 画像本体のオフセット */

int   headerPoint = 0;        /* ファイルから読み込んだバイト数 */

/*
 *   void readNextA( FILE *fp, int n )
 *   - ファイルの先頭からnバイトめまでを読む(絶対指定)
 *     すでに読み込んである部分はファイルから読まない
 */

void  readNextA( fp, n )
FILE  *fp;
int   n;
{
    if ( headerPoint >= n ) return;
    fread( &header[headerPoint], n - headerPoint, 1, fp );
    headerPoint = n;

    return;


/*
 *   void readNextR( FILE *fp, int n )
 *   - 現在読み込んでいる箇所からnバイト読む(相対指定)
 */

void  readNextR( fp, n )
FILE  *fp;
int   n;
{
    if ( n <= 0 ) return;
    fread( &header[headerPoint], n, 1, fp );
    headerPoint += n;

    return;
}

/*
 *   int readHeader( FILE *fp )
 *   - ヘッダを読み込む
 *     成功したら0，失敗したら-1を返す
 */

int   readHeader( fp )
FILE  *fp;
{
    int   t, n, i, j, k;
    int   t1, t2, tV;

    readNextA( fp, 8 );
    /* IDとバージョン */
    if ( header[0] != 0x4D ) return ( -1 );
    if ( header[1] != 0x4D ) return ( -1 );
    if ( header[2] != 0x00 ) return ( -1 );
    if ( header[3] != 0x2A ) return ( -1 );
    /* タグへのオフセット */
    t = ((((((header[4]<<8)|header[5])<<8)|header[6])<<8)|header[7]);
    readNextA( fp, t );
    /* タグの数 */
    t = headerPoint;
    readNextR( fp, 2 );
    n = ((header[t]<<8)|header[t+1]);
    /* タグ */
    t = headerPoint;
    readNextR( fp, n*12 );
    for ( i = 0; i < n; i++ ) {
        j = t + i*12;
        t1 = ((((header[j  ]<<8)|header[j+1])<<8)|header[j+ 2])<<8)|header[j+ 3];
        t2 = ((((header[j+4]<<8)|header[j+5])<<8)|header[j+ 6])<<8)|header[j+ 7];
        tV = ((((header[j+8]<<8)|header[j+9])<<8)|header[j+10])<<8)|header[j+11];
        if ( t1 == 0x01000003 && t2 == 0x00000001 ) {
            /* 画像の横幅 */
            imageWidth = tV>>16;
        } else if ( t1 == 0x01010003 && t2 == 0x00000001 ) {
            /* 画像の縦の長さ */
            imageLength = tV>>16;
```

```c
        } else if ( t1 == 0x01020003 && t2 == 0x00000001 ) {
            /* 1ピクセルのビット数(モノクロ) */
            bitPerPixel = tV >> 16;
            if ( bitPerPixel != 8 ) return ( -1 );
        } else if ( t1 == 0x01020003 && t2 == 0x00000003 ) {
            /* 1ピクセルのビット数(カラー) */
            k = tV;
            readNextA( fp, k );
            readNextR( fp, 6 );
            if ( header[k+1] != 8 ) return ( -1 );
            if ( header[k+3] != 8 ) return ( -1 );
            if ( header[k+5] != 8 ) return ( -1 );
            bitPerPixel = 24;
        } else if ( t1 == 0x01030003 && t2 == 0x00000001 ) {
            /* 圧縮形式(非圧縮のみサポート) */
            compression = tV >> 16;
            if ( compression != 1 ) return ( -1 );
        } else if ( t1 == 0x01060003 && t2 == 0x00000001 ) {
            /* 色モード(パレットあり/なし) */
            photometricInterpretation = tV >> 16;
        } else if ( t1 == 0x01110004 && t2 == 0x00000001 ) {
            /* 画像本体のオフセット */
            stripOffset = tV;
        }
    }
    /* 必要なタグを全部得たか */
    if ( imageWidth == 0 ) return ( -1 );
    if ( imageLength == 0 ) return ( -1 );
    if ( bitPerPixel == -1 ) return ( -1 );
    if ( compression == -1 ) return ( -1 );
    if ( photometricInterpretation == -1 ) return ( -1 );
    if ( stripOffset == -1 ) return ( -1 );
    /* ピクセルのビット数と色モードは合っているか */
    if ( bitPerPixel ==  8 && photometricInterpretation != 1 ) return ( -1 );
    if ( bitPerPixel == 24 && photometricInterpretation != 2 ) return ( -1 );
    /* 画像本体の先頭まで読み飛ばす */
    readNextA( fp, stripOffset );

    return ( 0 );
}

/*
 *   void makeResizeTable( int w0, int w1 )
 *   - 画像の横方向の拡大縮小を高速に行うためのテーブルを作る
 *     Bresenhamアルゴリズムの応用
 */

void  makeResizeTable( w0, w1 )
int   w0, w1;
{
    int   x, y, e, dx2, dy2;

    /*
     *  (0,0)-(w1-1,w0-1)の線分を発生する
     *     Bresenhamアルゴリズム
     */
    dy2 = (w0-1)*2;
    dx2 = (w1-1)*2;
    e = -(w1-1);
    y = 0;
    for ( x = 0; x < w1; x++ ) {
        resizeTable[x] = y;
        e += dy2;
        while ( e >= 0 ) {
            y++;
            e -= dx2;
        }
    }
    return;
}

/*
 *   void resize( w0, h0, x, y, w1, h1 )
 *   - 画像の表示サイズを決定する
 *     画面内に全画像が収まるように拡大縮小する
 *   (入力)
 *     w0×h0: Macintosh上での画像のサイズ
 *   (出力)
 *     (x,y):  X68000で画像を表示する位置
 *     w1×h1: X68000で表示する画像のサイズ
 */

/* X68000の512×512ドットモードの設定 */

#define   ASPECTRATIO 0.8   /* ピクセルの縦横比 */
#define   W_X68K      512   /* スクリーンの横のドット数 */
#define   H_X68K      512   /* スクリーンの縦のドット数 */
#define   HW_X68K     1.0   /* H_X68K/W_X68K */

void  resize( w0, h0, x, y, w1, h1 )
int   w0, h0, *x, *y, *w1, *h1;
{
    double   w, h, hw;

    w = w0 * ASPECTRATIO;
    h = h0;
    hw = h / w;
    if ( hw > 1.1*HW_X68K ) {               /* 画面サイズより縦長 */
        *h1 = H_X68K;
        *w1 = w * W_X68K / h;
        *x = ( W_X68K - (*w1) )/2;
        *y = 0;
    } else if ( hw < 0.9*HW_X68K ) {    /* 画面サイズより横長 */
        *w1 = W_X68K;
        *h1 = h * H_X68K / w;
        *x = 0;
        *y = ( H_X68K - (*h1) )/2;
    } else {                                /* ほぼ画面サイズ */
        *w1 = W_X68K;
        *h1 = H_X68K;
        *x = 0;
        *y = 0;
    }
    makeResizeTable( w0, *w1 );             /* 拡大縮小用テーブルを作る */
    return;
}

/*
 *   void readMonoImage( FILE *fp )
 *   - モノクロ画像をロードする
 */

void   readMonoImage( fp )
FILE   *fp;
{
    int      i;
    int      actX, actY, actW, actH;
    int      x, y, e, dx2, dy2;
    static unsigned short   pal[256];

    resize( imageWidth, imageLength, &actX, &actY, &actW, &actH );
    /*printf( "(%dx%d) -> (%dx%d)¥n", imageWidth, imageLength, actW, actH );*/
    /*fill( actX, actY, actX+actW-1, actY+actH-1, 63 );*/

    for ( i = 0; i < 256; i++ ) {
        pal[i] = ((i/8)<<11)|((i/8)<<6)|((i/8)<<1)|((i/1)&1);
    }
    /*
     *  縦方向の拡大縮小を行う
     *  - (0,0)-(actH-1,imageLength-1)の線分を発生するBresenhamアルゴリズム
     */
    dy2 = (imageLength-1)*2;
    dx2 = (actH-1)*2;
    /*e = -(imageLength-1);*/
    e = (imageLength-1);
    y = 0;
    for ( x = 0; x < actH; x++ ) {
        while ( e >= 0 ) {
            fread( mBuf, imageWidth, 1, fp );
            for ( i = 0; i < actW; i++ ) {
                rastBuf[i] = pal[ mBuf[resizeTable[i]] ];
            }
            y++;
            e -= dx2;
        }
        put( actX, actY+x, actX+actW-1, actY+x, rastBuf, actW*sizeof(short) );
        e += dy2;
    }
    return;
}

/*
 *   void readColorImage( FILE *fp )
 *   - カラー画像をロードする
 */

void   readColorImage( fp )
FILE   *fp;
{
    int   i;
    int   actX, actY, actW, actH;
    int   x, y, e, dx2, dy2;

    resize( imageWidth, imageLength, &actX, &actY, &actW, &actH );
    /*printf( "(%dx%d) -> (%dx%d)¥n", imageWidth, imageLength, actW, actH );*/
    /*fill( actX, actY, actX+actW-1, actY+actH-1, 63 );*/
    /*
     *  縦方向の拡大縮小を行う
     *  - (0,0)-(actH-1,imageLength-1)の線分を発生するBresenhamアルゴリズム
     */
    dy2 = (imageLength-1)*2;
    dx2 = (actH-1)*2;
    /*e = -(imageLength-1);*/
    e = (imageLength-1).
    y = 0;
    for ( x = 0; x < actH; x++ ) {
        while ( e >= 0 ) {
            fread( rgbBuf, imageWidth*3, 1, fp );
            for ( i = 0; i < actW; i++ ) {
                rastBuf[i] = ((rgbBuf[resizeTable[i]][1]/8)<<11)|
                             ((rgbBuf[resizeTable[i]][0]/8)<<6)|
                             ((rgbBuf[resizeTable[i]][2]/8)<<1);
            }
            y++;
            e -= dx2;
        }
        put( actX, actY+x, actX+actW-1, actY+x, rastBuf, actW*sizeof(short) );
        e += dy2;
    }
    return;
}

void   main( argc, argv )
int    argc;
char   *argv[];
{
    FILE   *fp;

    if ( argc < 2 ) {
        fprintf( stderr, "使用法: %s ファイル名¥n", argv[0] );
        return;
    }

    fp = fopen( argv[1], "rb" );
    if ( fp == (FILE *)NULL ) {
        fprintf( stderr, "%s: 指定されたファイルはありません¥n", argv[0] );
        return;
    }

    headerPoint = 0;
    if ( readHeader( fp ) != 0 ) {
        fprintf( stderr, "%s: サポートしていない形式のファイルです¥n", argv[0] );
        return;
    }

    screen( 1,3,1,1 );
    if ( bitPerPixel == 8 ) readMonoImage( fp );
        else    readColorImage( fp );

    fclose( fp );
}
```

[特集]

DTMへの招待



CONTENTS

概論
MIDIをめぐる環境'92……………………………………中野修一

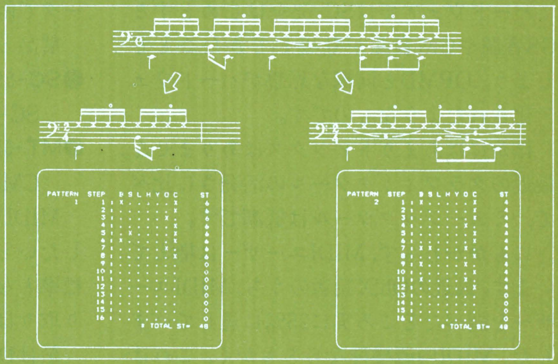
拡張された内蔵音源を使う
Z-MUSICとPCM8 ……………………………………西川善司

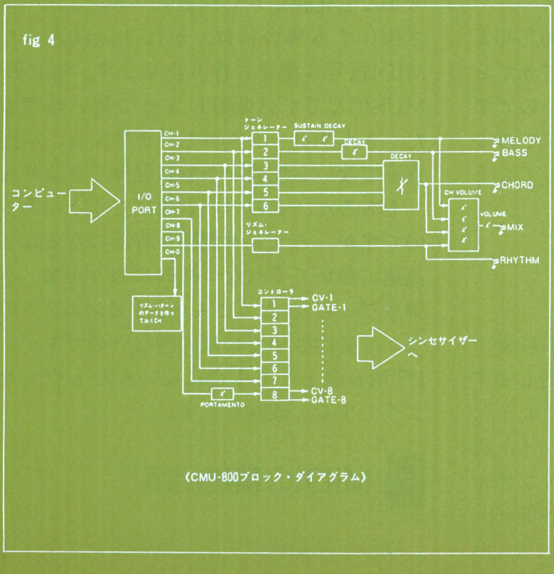
Z-MUSICによる
ローランド音源の基礎知識………………………………瀧　康史

市販アプリケーションを使う
MUSIC PRO-68K[MIDI]と仲よくなろう ………高橋哲史

Z-MUZIC用MMLデータプリプロセッサ
ZPP.X……………………………………………………岡本正和

おまけ
Z-MUSICでバビンチョ(ゲーム応用編) ……………西川善司





(CMU-800ブロック・ダイアグラム)

DTM——Desk Top Music.
コンピュータによる自動演奏というテーマはずいぶん古くから取り上げられていたものだ。音楽にはさまざまな壁がある。世の中，誰もが高度な演奏技術を持っているわけではないし，持っていたとしても，それは独奏曲に限られ，ひとりでは演奏不可能な曲のほうが圧倒的に多いのはいうまでもない。
やがて，シンセサイザとシーケンサはその壁を超えた。シンセサイザはよりリアルになり，電子楽器の標準インタフェイスとしてMIDIが生まれた。シーケンサにコンピュータが加わって，可能性は大きく拡大していった。そして，たったひとりでオーケストラに匹敵する演奏をすることが可能になった。
内蔵音源，そしてMIDI，そしてそフトウェア。顧みれば，そのための環境はすでに与えられていることに気がつくだろう。
すでに実現しているものとそうでないもの。「DTM」と「コンピュータミュージック」という言葉のあいだの微妙な違い。それが，我々の現在の音楽環境を象徴しているのではないだろうか。

概論

MIDIをめぐる環境'92

Nakano Shuichi 中野 修一

現在の私たちにとっての，MIDIとコンピュータミュージックをめぐるさまざまな問題についてまとめてみましょう。また，初心者のための基本MIDI用語解説もあわせて行います。

DTM前夜

音楽特集のたびに思うことですが，最近2年間でどの程度環境が変化したかというと……。ソフトウェアではやはりZ-MUSICを発表したことでしょうか。MusicstudioはMu-1 Superに変わりましたが，MUSIC PRO-68Kは相変わらずですし，OPMDRV2/3は日の当たる場所に出そうにありません。X68000ではMIDI対応のゲームは一般化し，最近では同人ソフトまでがMIDIに対応しています。

ハードウェアではSC-55をはじめとするGS音源やGM音源が登場してきたことでしょう。OPMDと同様な思想がハードウェア化されたようなものです。

ほかにはあまりトピックスはありません。最近のグラフィックツールの活発さに比べて，ミュージックツールは低調です。

にもかかわらず，MIDIユーザーは増えています。最近のOh!X読者のうち，MIDIユーザーを調べたところ実に25%に達していました。MIDI内蔵機種ならともかく，これは驚異的な数字です。

このような現状では「DTM」という言葉からはどうも吉野屋でハンバーガーを見るような違和感が感じられます。X68000をめぐる環境を見ても，ちょっと見てもよくわからないMMLを使って呪文のようにデータをコーディングする輩には似合わない単語です。X68000で作られた素晴らしいデータというのも多数存在します。しかしそれらのデータからはどこか汗と血の匂いがします。

偏見を隠さずに表現するならば，現在のDTMとは，使いやすいかどうかは別にして見栄えのいい画面上で聞くからに「コンピュータミュージック」というデータを作る人たちの世界，というイメージがあります。

ですから，「DTMへの招待」というのが意味するのは，すでにできあがった現在のDTMへの招待ではなく，一緒に「本当のDTM」自体を作るためのものであることは間違いありません。

音源の選び方

こと，DTMということで音源を選ぶなら，おすすめできるものはローランドのSC-55，CM-300，CM-500，CM-32L，CM-64といった音源モジュールになります。要するにMT-32系列とSC-55系列であれば，データで困ることはないからです。

製品としては次のようになります。

●SC-55系列

SC-155，CM-300，CM-500，JV-30

●MT-32系列

CM-32L，CM-64，CM-500

MIDIを使うならいろいろな楽器を使用したいというのは当然な話です。しかし，標準となるものがなにかないとデータのやり取りはかえって不便になります。

敵はDTMですから，なんらかのプラットホームの確保が必要です。

最右翼はSC-55でしょう。CM-300/500よりもインジケータ，マニュアルのできでおすすめです。TG100でもかまわないのですが，GMよりもGS規格のほうが上位であること，すでにSC-55がかなり普及してしまっていることが強みです。

これらDTM用の音源には，

動作速度が速い

という特徴があります。

MIDIデータは31250bpsで転送されます。これは秒間3125バイトの情報に相当します。全音符の分解能を192とした場合でもテンポ120時に必要な情報量は秒間288バイト程度の計算になります。これは1チャンネル分の値ですから16チャンネルだと秒間4608バイトという限界を超えたデータが送られてくる可能性があるのです。

無論，MIDIでこれだけのデータを扱うことは不可能ですが，MIDIフォーマット自体がデータの量に対してかなりサバを読んだ作りになっているのがわかると思います。

楽器のほうはもっとサバを読んでいます。困ったことにMIDI転送速度内であっても，データが混雑してくると受け取ってくれない音源が多いのです。というより，ちゃんとMIDI転送速度に追いつく音源というのは見たことがありません。MIDIの規格ではすべて音源側で責任を持って受け取らねばならないことになっているのですが，楽器の都合により「重い処理」というのはあるみたいです。

なぜか鳴らない音がある……という場合や違う音色が鳴っているという場合の原因は，ほとんどが楽器側が追いついていないことによるものです。複数のトラックで同時に音色を切り換えたりするとよくこのようなことになります(対策？タイミングをずらすしかありません)。PCM系楽器の音色切り換えは一般的にいって重いようです。

楽器の処理が速いと凝った操作をしてもちゃんと発音してくれます。コンピュータで操作する場合には，かなり込み入ったMIDI信号も簡単に作り出せます。特にZ-MUSICなどでは楽器にとって重い処理もたくさんサポートされていますから，高速音源でないと性能を十分に発揮できません。

もちろん，音のいい音源はたくさんあるのですが，どうも大規模な音源になるほど高速に動いてくれない傾向にあります。手弾きだったり，たくさんのMIDIシステムのなかで動作させるには問題ないのですが，DTM音源として1台になにもかもやらせようとすると破綻します。

MIDIの基礎用語

楽器を外部制御するためのインタフェイ

スとして作られたのがMIDI規格です。コンピュータを使用することで，MIDIは最高のパフォーマンスを発揮します。ここではMIDIのキーワードを見てみましょう。

●チャンネル数と発音数

基本的には1チャンネルでたくさんの音を鳴らすことができます。ローランドに限らず，最大発音数は実際の発音数と一致しないことが多々あります。

24パーシャルのSC-55で2パーシャルの音色を多用した場合，13チャンネル以上使うのは無駄なことです。

発音の優先順位がチャンネルによって規定されていることもあるので（GSなど），コーディングの際に，消えると困る音はそれなりの配慮をしておくべきでしょう。

●パンポット

音のステレオ定位です。128段階の指定ができますが，どう解釈されるかは楽器によって違います。MT-32系では左右逆というのも有名です。

●コントロールチェンジ

コントロールチェンジは演奏表現を広げるための手段として用意されているもので，実にさまざまなものがあります。どのパラメータがどんな機能に対応するかということはMIDI協議会で決められていますが，各楽器でどれに対応しているかが異なってきます。別にすべてに対応する義務はないわけです。

手元にある楽器のインプリメンテーションチャートからコントロールチェンジを抜き出してみました。

0　バンクセレクト（MSB）
1　ピッチモジュレーション
2　ティンバーモジュレーション
5　ポルタメントタイム
6　データエントリー（MSB）
7　パートボリューム
10　パンポット
11　エクスプレッション
12　エフェクト1
13　エフェクト2
32　バンクセレクト（LSB）
38　データエントリー（LSB）
64　ダンパー（ホールド）
65　ポルタメントスイッチ
66　ソステヌート
67　ソフト
91　エフェクト1
92　エフェクト2
93　エフェクト3
96　データインクリメント
97　データデクリメント
98　NRPN(LSB)
99　NRPN(MSB)
100　RPN（LSB）
101　RPN（MSB）
120　オールノートオフ
121　リセットオールコントローラ

音色バンクセレクトが追加された経緯があるので，0～127のすべてが埋まっているわけではないと思います。

これらの信号の特徴はリアルタイムに反映されるということでしょう。エクスクルーシブを使って，FM音源のようにリアルタイムに音色パラメータを書き換えようとしても，多くのMIDI楽器は反応しません。コントロールチェンジによるものなら，その場で反応するので非常に微妙な制御も可能です。Z-MUSICではARCCなどで音源の性能をめいっぱい引き出すことができます。コントロールチェンジは動作も軽く，積極的に使っていい命令といえます。

●エクスクルーシブ

コントロールチェンジが各機種で共通に使える制御コードなのに対して，特定機種だけで有効なのがエクスクルーシブメッセージです。機器の設定に関するものが多いのですが，コントロールチェンジが弱いときにはエクスクルーシブで対処する場合もあります。

MT-32などではリバーブを変更するためにエクスクルーシブメッセージを送らなければなりません。M1ではパンポットを変えるためにエクスクルーシブを使わなければなりません。

●ボリュームとベロシティ

Z-MUSICでは音量はトラックボリューム（@Vコマンド）で与えます。

コントロールチェンジにエクスプレッションというものがあります。これはほぼボリュームと同義で使用できます。エクスプレッションの存在意義はいまひとつわからないのですが，音色によってはボリューム値で音の性格を変えるものがあるので，おそらくそれに影響しない音量調節として使うのでしょう。

同様なものにベロシティ（@Uコマンド）があります。ベロシティはMIDIにおける「正式な」音量なのですが，なぜかZ-MUSICではボリュームに一歩譲っています。ベロシティは音量のみならず音質も変化させる指定として知られています。ベロシティは音の「強弱」を，ボリュームは音の「大小」を調整します。

コンパイル時にデータが確定し（ボリュームは演奏時まで不確定），しかもノートオン情報として送られるのでコントロールチェンジによるボリュームより負担が少なくなっています。

'93へ向けて

最近はDTM用の音源でも結構本格的な音楽が作れるようになりました。

進化したグラフィック環境が絵の描ける人の能力を拡大したのに対して，進化した音楽環境は演奏に関する限りではプロとアマチュアの差を縮めつつあります。プロミュージシャンと同じ楽器を使うことも不可能ではありません。表現が楽器に依存する部分が大きく，自由度がそれほど大きくないことから，同等の演奏を行うことも夢ではないといえるでしょう（打ち込みに関しては）。

DTMもそろそろ「音楽をコンピュータでやってみました」というレベルから脱却すべきです。DTMが録音された媒体と本質的に違う点も忘れられているような気がします。常に「生演奏である」という特性を最大限に生かすなら，データの作成や演奏だけでなく，アレンジがその場でできるような環境こそが重要になってくるでしょう。

＊　＊　＊

冒頭でも述べたように，X68000における現在のミュージックツール環境はまだ十分ではありません。

分野別で見ると，

リアルタイム入力：十分でしょう
MML入力：十分でしょう
ステップ入力：拡充されるべきです
楽譜入力：改善されるべきです
楽譜作成：貧弱です

リアルタイム入力ではMu-1 Superがあります。機能，性能ともに優秀なソフトです。MMLではZ-MUSIC，楽譜関係はMUSIC PRO-68Kです。しかしMUSIC PRO-68Kは基本的に5年前のソフトですので改善すべき点が目立ちます。

なににもまして全体の統合度が低いことが問題ではないでしょうか。

現在，Z-MUSICは環境を再構築しています。次期リリースバージョンに予定されているZMUSIC.X ver.1.50において当面仕様は凍結され，アプリケーションの拡充に重点を移していきます。SX-WINDOW開発キットの状況をにらみ，行動を起こそうと待ち構えている一派もいるようです。

DTM前夜は次第にあわただしさを加えつつあります。

拡張された内蔵音源を使う

Z-MUSICとPCM8

Nishikawa Zenji 西川 善司

なにもMIDIだけがDTMというわけではありません。PCM8の登場は内蔵音源の可能性を大きく広げました。ここではZ-MUZIC ver.1.10でサポートされたPCM8対応機能を紹介します。

PCM8.X

X68000は本体の内蔵音源だけでも，かなりの表現力を持ったパソコンです。OPMというFM音源LSIによりちょっとした楽器並みの8音の発音が可能でした。

もうひとつの音源であるAD PCMはパソコンをしゃべらせるためのものという認識が浸透していました。

AD PCMを音楽演奏に用いるという，いわば大道芸的な発想から，X68000の音楽に革命をもたらしたのがボスコニアンです。

通常の演奏データにAD PCMを同期させるOPMA/OPMDが発表され，瞬く間にAD PCMは演奏用音源として一般化していきました。

従来FM音源で担当していたドラム部分が演奏用に開放されたことにより，FM音源のチャンネルにも余裕が生まれてきます。FM音源が究められていくにしたがって，音楽データもいっそうの進歩を遂げていきました。

AD PCMによりリアルなドラムサウンドを実現したものの，ハードウェアの制約上ドラムの発音数は1音に制限されています。このため，あらかじめ合成しておいた音色を使うといった手法なども登場しましたが，それでも打楽器の余韻などは表現できませんでした。

そして，……。

＊　＊　＊

本誌6月号の付録ディスクに収録されたPCM8.X（江藤啓氏作，以下PCM8）はX68000本体をいっさい改造せずに高度なソフトウェア処理でAD PCMの発声数を8倍に拡張してしまうスーパーアプリケーションです。この機能を音楽演奏に生かさない手はない，ということでZ-MUSICも早急にこれに対応しました。ここでは，このPCM8のZ-MUSIC上での具体的な使い方について解説します。

すでにいくつかのPCM8対応ミュージックデータがOh!X LIVEで発表されていますので，それらを参考にしながら読み進めてください。

Z-MUSICの組み込み

7月号に示したようにZ-MUSIC ver.1.10ではPCM8に対して2通りの対応の方法を持っています。

●ポリモード

まず，PCM8を先に組み込み，そのあとオプションスイッチ'-O'を添付してZ-MUSICを組み込みます。これで従来のAD PCM 1声のZMS・ZMD演奏データをPCM8を用いてポリフォニックに演奏するモードになります。

X68000のAD PCM音源はもともと単音ですから，従来の演奏データでは，たとえばシンバルを叩いたあと，このシンバルが鳴り終わらないうちにスネアを叩いたとする，ると，AD PCMの単音発声の制約からシンバルの音がブツリと切れてスネアの音に切り替わっていました。ここをブツ切りにしないで（PCM8.Xを用いて）ちゃんと前後の音を重ねて演奏してしまおうというのがこのモードです。

このモードをZ-MUSICではPCM8ポリモードと呼ぶことにします。

特に，OPMDなどで作成された音楽をPCM8で演奏したいという場合に指定すると効果があります。最初からZ-MUSICでデータを作る場合は次に説明する独立チャンネルモードを使用したほうがよいでしょう。

組み込み例

A>PCM8

Z-MUSICの今後

非常にお待たせしている改訂版MOOKですが，リリースされるのはZMUSIC.X ver.1.50となる予定です。ver.1.01からver.1.10への変更点だけでもかなりの量ですが，ver.1.10からver.1.50への変更点もそうとうのものです。ここではその概要を紹介しておきましょう。

基本的には，こまごましたところが拡充され，支援プログラムが作りやすくなり，ミュージックデータ作成時にver.1.01で見られたさまざまな但し書きによる制限が取り払われています。

FM音源のダンパー機能もかなりまともになり，デフォルト調号の指定ができるようになり，メッセージ表示で画面を壊すこともなくなりました。

PCMにバンクがサポートされ，たとえばPCM8で複数の非ドラム系楽器をAD PCMで鳴らす際にも音階割り当てが競合しなくなります。

モジュレーション関係は，ver.1.10でアンプリチュードモジュレーションに対応しましたが，さらにモジュレーションで三角波以外の波形が使えるようになります。一般的な鋸波と矩形波のみならず，ユーザー定義の自由なパターンが使える波形メモリ機能が拡充されます。波形メモリでは，位相反転や逆スキャン，指定位置からのループなどが可能です。

これはFM音源だけでなくMIDIにも適用できます。それに伴い，ピッチモジュレーションを楽器のモジュレーション機能を使わずに処理する，いわば「力技モジュレーション」を実現しています。

また，波形メモリはARCC（Assignable Real-time Control Change）にも適用できるので，MIDIにおける表現力はとてつもないものになります。当初，「あまり使わないし，速度が……，サイズが……」といっていたわりには，このテの機能に関しては最強の仕様になっています。

今回のバージョンアップでの最大の影響は，ワークの見直しでしょう。代表的なものを除いてワークエリアの変更が行われています。機能拡張のため，ZMDコードも一部互換性がありません。システムコールはかなり増えています。データの互換性はZMSレベルでは確保されています。しかし，ワークの変更のため一部の支援プログラムでは変更が必要になってくることもあります。支援プログラム開発中の方で思い当たる人は編集室まで連絡してください。

A>ZMUSIC －O －P200 －T100

●独立チャンネルモード

PCM8を組み込み，そのあと通常どおりにZ-MUSICを組み込みます。これでPCM8が管理する疑似的な8つのAD PCMチャンネルを個別に使えるモードになります。

この時点でZ-MUSICでシーケンス可能な音源構成は，

FM音源	8チャンネル
AD PCM音源	8チャンネル
MIDI音源	16チャンネル

の32チャンネル分になります。

6月号以降でOh!X LIVEコーナーに掲載されたPCM8対応の曲はすべてこのモードで動作します。

このモードを指定するとAD PCMトラックでも音量指定やキーオン/キーオフ,周波数（5段階）は完全に独立にコントロールできます。しかし，独立に制御できるといっても，ハード的な制約によりパンポット（音場）は各チャンネル独立には機能しません。

組み込み例

A>PCM8
A>ZMUSIC －P200 －T100

独立チャンネルモード

●チャンネルアサイン

AD PCMを指すチャンネル番号を異なったトラックにアサインすることによって，それぞれのトラックが順番にADPCM1チャンネル，ADPCM2チャンネル，……，ADPCM8チャンネルのように割り当てられていきます。

X-BASIC（MUSICZ.FNC）では，たとえばトラック9～16をADPCM1～8チャンネルに割り当てたい場合は，

m_ch("FM")のとき

```
    for i=9 to 16
        m_alloc(i, 1000)
        m_assign(9, i)
    next
```

m_ch("MIDI")のとき

```
    for i=9 to 16
        m_alloc(i, 1000)
        m_assign(25, i)
    next
```

あるいはm_assign2()を用いて，

```
    for i=9 to 16
        m_alloc(i, 1000)
        m_assign2("ADPCM", i)
    next
```

のようにします。

ZMS書式では，

m_ch("FM")のとき，

```
    (M9, 1000)(A9, 9)
    (M10, 1000)(A9, 10)
    (M11, 1000)(A9, 11)
    (M12, 1000)(A9, 12)
    (M13, 1000)(A9, 13)
    (M14, 1000)(A9, 14)
    (M15, 1000)(A9, 15)
    (M16, 1000)(A9, 16)
```

m_ch("MIDI")のとき，

```
    (M9, 1000)(A25, 9)
    (M10, 1000)(A25, 10)
    (M11, 1000)(A25, 11)
    (M12, 1000)(A25, 12)
    (M13, 1000)(A25, 13)
    (M14, 1000)(A25, 14)
    (M15, 1000)(A25, 15)
    (M16, 1000)(A25, 16)
```

または，

```
    (M9, 1000)(A ADPCM, 9)
    (M10, 1000)(A ADPCM, 10)
    (M11, 1000)(A ADPCM, 11)
    (M12, 1000)(A ADPCM, 12)
    (M13, 1000)(A ADPCM, 13)
    (M14, 1000)(A ADPCM, 14)
    (M15, 1000)(A ADPCM, 15)
    (M16, 1000)(A ADPCM, 16)
```

のようにします。

●使用解禁のMML

PCM8使用時でも，従来AD PCMトラックで使用可能だったMMLはすべて使用可能です。PCM8独立チャンネルモードでは従来AD PCMトラックで使用の禁止されていたV，@Vのボリュームコマンドが使用可能になります。ボリュームは0～16が有効範囲で原音量は9です。@V使用時は0～16へ換算されます。

●機能拡張されるMML

独立チャンネルモードでは周波数設定コマンドである@F命令は各トラック独立に機能するようになります。

また，PCM8が扱うことのできる新データ方式「16ビットPCMデータ」「8ビットPCMデータ」に対応しました（囲み記事参照）。それぞれ，

@F5　16ビットPCMデータ方式
@F6　8ビットPCMデータ方式

で対応します。

メモリの消費量は増大しますが，CPUの負担をどうしても軽くしたいという場合は，AD PCMデータをリニアPCMデータに変換して用いるとよいでしょう。

その他

Z-MUSICの音楽演奏も割り込み処理，またPCM8の発音処理も割り込み処理です。よって，10MHzのX68000だとお互いの割り込み処理がフル回転（？）したときには音楽のテンポに影響が出る場合があります。また，PCM8対応曲を演奏中にCPU負荷が高まるのは仕様上どうしようもありません。ご了承ください。

PCM8によってX68000内蔵音源の新たな可能性が切り開かれました。このような素晴らしいアプリケーションを開発し，そして快く公開された江藤啓氏に敬意を表します。

Oh!XではPCM8とZ-MUSICを使用したミュージックプログラムの投稿を募集します。ドシドシ投稿してきてください。

8,16ビットPCMデータ方式とは?

X68000の扱うAD PCMデータとは，いわばサンプリングしてできたPCMデータを一種の圧縮アルゴリズムで圧縮したものと考えられます。音量の変更や，周波数変換，合成処理を行うにはこのAD PCMデータ方式からPCMデータ方式へ変換しなければなりません。PCM8ではこの変換処理を割り込みでリアルタイムで行っていますが，もしこの処理を省けるならば，かなり割り込み処理の負担を軽くすることができるはずです。そこでPCM8では無圧縮のPCMデータを演奏できるようになっています。

16ビットPCMデータ方式は符号付き16ビット整数により表現されたデータです。同一音長のAD PCMデータの4倍のデータ長になります。ちょうどZVT.Xで'－C'スイッチで生成されるPCMデータと同じものです。ひとつのデータが2バイト構成であることからデータ長は必ず偶数でなければ正常に鳴りません。

8ビットPCMデータ方式は同様に符号付き8ビット整数により表現されたデータです（－128～127）。同一音長のAD PCMデータの2倍のデータ長になります。現在のところ，8ビットPCMを扱うツールはありませんので，16ビットデータからプログラムで作成するなどしてみてください。

なお，8ビットPCMデータとしてはAMIGAのAudio IFFファイルのBODY部分がこの仕様に当てはまります。FM TOWNSのPCM音源は8ビットPCMですが，7ビット＋符号1ビットという仕様のため(－127～127)，TOWNSのPCMデータをそのまま扱うことはできません。

8ビットPCMデータ/16ビットPCMデータともに特別な設定なしにZPCNVでZPDデータを作成できます。もちろんAD PCMデータと混在させてZPDデータにすることもできます。

再生周波数はどちらとも15.6kHz固定ですが，AD PCM方式との混在が可能でAD PCMデータ側はこのときも5段階の周波数が使用可能です。

Z-MUSICによる

ローランド音源の基礎知識

Taki Yasushi 瀧 康史

現在，パソコンにおけるコンピュータミュージックの標準音源となっているMT-32，そして新たな標準となろうとしているSC-55。これらローランド製の音源について，その特徴と使用上の注意をまとめてみました。

私の持ってる楽器でいちばんの愛機，それはもちろん，ヤマハのアップライトピアノです(笑)。2番目にD-70。3番目にCM-64かな？　どちらもローランド製品ですけど，別にローランドの楽器のファンというわけでもなし，D-70は夜のピアノの練習用として買っただけだし（本当はクラビノーバを買おうとしたけど，友人に大反対されたの），ピアノの音で決めたんだよね。

それからCM-64を私が買ったのは，別に仕事柄とか実はそんなんじゃなくて，ゲームをワイドな音で楽しみたかったからでした。きっと，サコムのジェミニウイングがCM-64に対応してなかったら，CM-32Lを買ってたことでしょう。

とりあえず，現状ではDTM音源としてローランド製品（特にCM-32L，CM-64，SC-55)が隆盛を極めています。皆さんのなかにもローランド製品を使っている人が多いはずですよね。とはいっても，実はゲーム専用の音源になっている人もけっこういるんじゃないでしょうか。そういう，もったいない人のための音源活用講座をしてみましょう。

CM-64だけを使う

Z-MUSICってCM-64(のうちのRSPCM：CM-32Pの部分）をコントロールする命令がないんですよね。それにはエクスクルーシブデータの転送をすればいいのです。一見面倒くさそうに見えても，1回やってしまえばあとはわりと簡単に覚えてしまいます。エクスクルーシブがなにかなんて覚える必要はありません。

Z-MUSICには専用命令.ROLAND_EXCLUSIVEが用意されているので，面倒なチェックサム計算などは必要ありません。

```
.ROLAND_EXCLUSIVE 16, 22={
        アドレス
        データ群    }
```

という書式です。最初の「16，22」はCM-64のIDです。アドレスとデータはインプリメンテーションチャートを見てください。「アドレス」と「オフセットアドレス」に注意すれば特に問題はないでしょう。

CM-64で行われるエクスクルーシブデータ転送のほとんどはパーシャルリザーブとリバーブ操作です。CM-64という楽器は，パーシャルリザーブがなかなか不思議な配分にされていて，必ず一度はやらなくてはデータがまともに鳴らないのです（しかし私は手抜きのため連載のサンプルではやったことがありません)。まず，パーシャルリザーブから解説しましょう。

パーシャルはローランド音源で共通に使われる言葉で，音源の基本モジュールを意味します。1つひとつが独立したシンセサイザでそれぞれで音を発声できるのですが，楽器として扱う場合に，いくつかのパーシャルをまとめてひとつの音色として扱えるようになっています。

たくさんのパーシャルを使えば複雑な音が出るけど同時発音数が少なくなる，というのは，きわめて当然のことでしょう。

CM-64は，起動時には次のようにパーシャルリザーブが配分されています（デフォルトチャンネル時)。

LA音源側（CM-32L，MT-32)

MIDI ch	2	3	4	5	6	7	8	9	10
Part	1	2	3	4	5	6	7	8	9
P Res.	2	10	6	4	3	0	0	0	6

計32パーシャル

PCM音源側（CM-32P)

MIDI ch	11	12	13	14	15	16
Part	1	2	3	4	5	6
P Res.	2	8	21	0	0	0

計31パーシャル

ただし，実際はLAもPCMも1音(パーシャル）だけで出力する音色は少なく，実際には2つの異なる音をなんらかの方法で混ぜあわせて鳴らしたり，もしくは同じ音をピッチをずらして2音鳴らしたりしているため，この値よりももっと少ないのです。標準にプリセットされている音色も当然この方法を使っていて，これらは音色表のPtl #というところに書かれています。

たとえば，LA音源のアコースティックピアノの音色番号は1，これのパーシャルはなんと4，32/4＝8。すなわち，4パーシャルの音を使ってしまうと，同時に8音しか鳴らないのです。

合計パーシャルの範囲内なら，どのチャンネルでも自由に発音することができるのですが，リリース中の音も含めて，パーシャル数が限界を超えると後着優先で前の音がぶつぶつ途切れていきます。ありがちなのが，データ上で発音数を計算しているのに音切れを起こすケース。データ上で発音をやめても実際にはリリース音が鳴っているわけだからきちんと発音数を管理していなくちゃいけません。

そうなってくると，なかには途切れてもかまわない音と困る音の相違とか，チャンネルによって使う音数の相違とかで，優先順位を決めたらよかろうってことになりますよね。その設定がパーシャルリザーブなのですよ。

Z-MUSICを使ってCM-64（PCM部）のパーシャルリザーブを変えるには，

```
.ROLAND_EXCLUSIVE 16,22={
  $52, 0, 4      /このアドレスから
   6, 5, 5, 5, 5, 5  }
```

のように最初の2行をおまじないにして，各チャンネルで使いたいパーシャル数を書いていくだけです（ただし，合計は31以下にすること)。

この設定値は，簡単にいえば重音する数の多い順に決めればいいのですが，リリース音も含まれることをよく考えなくちゃなりません。たとえば，Cから1オクターブ上のCまで，グリッサンドする場合，楽器が金管楽器なら，パーシャルは音色が要求

する数だけでいいのですが（パーシャル１の音なら１ね）ピアノのように，一度にたくさん音が鳴る楽器だと，グリッサンドの分だけ，すなわち，

8×音色が要求する数

が必要になってしまいます。

このようなとき，デフォルトのままパーシャルが設定されていたとして，ピアノがチャンネル２(デフォルトではパーシャル２)で，トランペットが３(パーシャルは１)では，ピアノの音はプチプチ切れてしまうのに，トランペットはなぜか重音してしまう，という事態が起きてしまいます。ということで，パーシャルリザーブは，

（使う楽器の同時に発生できる数）

×（音色の求めるパーシャル数）

から割り出すというのが理想です。

例1)　トランペット

生楽器の同時発声数：当然１

必要パーシャル(LA)：３

ゆえに理想パーシャルは３

例2)　ディストーションギター

生楽器の同時発声数：６(or ５)

必要パーシャル(PCM)：

CARD ７ の 51 なら２

ゆえに理想のパーシャルは6×2＝12

この例ではPCMの全体パーシャル31のうち，12も使ってしまうことになります。ディストーションが２本のツインギターなら，ギターだけで24パーシャルです。

この音色はパーシャル２つを使って，デチューンをかけているため，音に厚みがあるのですが，我慢してパーシャル使用数１のCARD7-43を使えばひとつですみます。

CM-64とほかの楽器を併用する

CM-64はひとつの楽器だけで２～16チャンネルを使ってしまう楽器です。ということは，ほかの楽器をつなぐ際に空いているところは１だけなので，そこにつなぐしかないんですよね。デフォルトでは。

しかし，最近はキーボードもマルチティンバー化してるので，実際はどこかのパートが同時に鳴ってしまうことになります。

そこで，使用するMIDIチャンネルを変えることから始めましょう。

まず，CM-64とD-70(５パート＋リズム１)を使うとして，MIDIチャンネルの配分は，D-70が１～5ch(５パート)，PCMが６～11ch（６パート)，LAが12～14ch（３パート)。D-70リズムが15ch，LAリズムが16chにしたいという設定で話を進めます。ちょっと複雑ですけどねぇ。

CM-64はLAパートに９（９番はリズム）RSPCMに６つのパートを持っています。このパートというのは，デフォルトでは２～16chで埋められていますが，変更も可能です（変更だけではなくマスクも可能です)。

D-70側のエクスクルーシブ転送で制御する方法を知っても嬉しくない人がほとんどでしょうから，こいつはフロントパネルの設定で，すでに１～5chおよび，リズムが15chになっているとしましょう。

さて，Z-MUSICはCM-32Lには対応しているので，LA音源側は簡単に定義できます。最初にこいつの設定をしておきましょう。

ここではLAのパートは１～３までしか使いません。４～８は未使用なのでマスク(OFF)します。マスクしないで省略すると，５～9chのデフォルト設定のままだから，注意するようにね。

具体的には，下の命令を実行します。

```
.MT32_PART_SETUP＝｛
    12,13,14,17,17,17,17,17,16｝
```

Z-MUSICのマニュアルを見ると，引数の値はMIDIチャンネルなので１～16，それ以外はマスク（OFF）と書いてあります。

PCM側は専用命令が用意されていないので，.ROLAND_EXCLUSIVE命令を使用します。

```
.ROLAND_EXCLUSIVE＝｛
              $52,00,$0a
              5,6,7,8,9,10｝
```

ここでの引数は，さっきとはちょっと違って，設定したいチャンネル－１を設定します。16ではマスク（OFF）にされます。

これだけの設定で，MIDIチャンネルが理想の状態に設定できました。LA音源は３パートしか鳴らせませんが，その分，パーシャルリザーブの設定が１チャンネルにつきたくさん割り当てられるので，いいことにしましょう。

CM-500ではGS＋LAで設定できるので，これらの設定は各自の検討事項としてください。もっとも，.SC55_PART_SETUPという命令が用意されているというので簡単にできるでしょうけど。

エフェクタの使い方

まずは，内蔵エフェクタの説明から。MT-32やCM-64ではリバーブ，SC-55ではリバーブとコーラスが内蔵されています。

チャンネルごとにエフェクト量が選べるSC-55はまだいいのですが，MT-32やCM-64ではチャンネルごとのON/OFFしか設定できませんので，使い方が多少制限されます。

SC-55ではリバーブとコーラスのタイプをそれぞれ設定して（これは各チャンネル共通)，チャンネルごとにかかり具合が調整できます。もちろん，リバーブとコーラスで独立に設定できます。

Z-MUSICではMT-32とSC-55に関しては，IDを登録することで，@Eコマンドが使用可能になります。

SC-55に関してはリバーブ，コーラスのパラメータをトラックごとに指定できます。

例)　　@E80，127

リバーブを80，コーラスを127に設定

MT-32の場合は，ON/OFFを楽器のパート（デフォルトチャンネル番号－１）に指示するという制限された使い方になります。

例)　　@E1,0

パート１のリバーブをOFFにする

どうして，ほかの機種用のものをサポートしてないかというと，作者が持ってなかったという理由と，ほかの機種ではエクスクルーシブ操作が重すぎて演奏中に切り換えることはあまりすすめられないからです。MT-32はかなり軽い音源ですし，SC-55はエクスクルーシブを使わずにエフェクト操作ができるのです。

TG100もコントロールチェンジでリバーブを操作できますが，GS音源と同じ番号なので，“@I$41,$42,$10”とGS用のIDを登録すれば制御できるはずです(Yを使うほうが無難か？)。

CM-64のPCMパートではエクスクルーシブでリバーブのパラメータを送らなくてはなりません。CM-64だと重すぎてまったく使えないかというと，そうでもありません。そのチャンネルの発音に余裕があるときにはエクスクルーシブデータを送っても大丈夫です。

CM-64で設定できるリバーブのタイプを見ると，次のようになっています。

●Room

リバーブというのは残響効果のことです。エフェクタそのものは，ある仮定した閉空間のなかで，壁の持つ音の反射率とか，音がどのように跳ね返ってきてそして集まるかを計算したものだそうですが，どうせそんなことまでやってないでしょう。

ということで，これはその閉空間が部屋ぐらいってことみたいね。

ツヤがある響きをするので，うまくタイムや，レベルを設定すればアクションのあ

る響きをさせることができます。

●Hall

これも文字どおり，ホールの持つ残響効果を持たせたもの。言葉に直せば，広がりのある響きってことかな。うにゃ～んとした効果ですね。FM音源で残響効果が憧れだった分，使っている人が多いようです。無難な設定なのでよしとしましょう。

●Plate

おそらく，名前（金属板）の意味どおり，リバーブで囲ってある素材を金属にしたものでしょう。

サウンド的には音がこもらないリバーブといった感じかな？　ピアノなんかにかけたらクリアな感じだったな。トランペットにかけたら，ペラペラした感じになってしまって，それでいてファンキーなので使ってみると面白いかもしれません。

●TapDelay

これは，さまざまな名前があります。同じローランドでも，D-70ではCrossDelayっていいます。あと，PanpotDelayとか。

要するに，Delayが回数を繰り返すごとに左右に飛ぶのね。まあ，特殊効果に近いものがあって，CM-64は全体でひとつしか使えないから，あまり使い道はないかもしれません。

＊　＊　＊

パラメータのタイムというのは，残響効果（リバーブ）や反響効果（ディレイ）のかかる時間を表しています。要するに，大きくするといつまでも，余韻が残るってこと。レベルは余韻を残すときの音量。これを大きくすると，いつまでたっても音が減衰しないってわけ。

とりあえず，これらのパラメータを駆使して，CM-64ではエフェクトを決めねばなりません。

無難な線は，デフォルトのままかな？　エフェクトには趣味が出てきますが，一般的にいって，あんまり強いと下品です。FM音源ではエフェクトはなかったから，思わずHall,Time 7,Level 7って，思いっきりかけちゃう人がいるけど，これもね。ずぅっと聞いてると，あんまりわぁわぁ残響が残るのもくどくなってしまいます。

私の場合，基本的には薄化粧（エフェクト効果のことを化粧にたとえる人は多い）なので，エフェクトがあるかないかわかんないけど，とっちゃうとわかるよって感じにかけます。

ジャンル的にも違いますが，たとえば，ポップスなら，

Room　Time 2　Level 4

クラシックなら，

Hall　Time 4　Level 6

室内音楽なら，

Room　Time 6　Level 3

こんな感じに，分けて使ってます。ポップスのほうは，メリハリの効く効果がほしいのでタイムを短めにとって，音に切れのよさを与えます。

クラシックなら大きめのホールで，室内音楽は広めのお部屋でのんびりとって感じかな。

それから，私はD-70とあわせて使うときは，たいてい1音しかLAで鳴らさないのでまだ実用的なんですが，

TapDelay　Time 5　Level 7

というのを，パイプスで使っています。パンパイプってことね。

でもこの場合，LAにほかのパートがあると腐るので，その部分のエフェクトを切らなくてはならないってことに問題がありますね。

まあ，リバーブのエフェクタは最近安いですし，FMにもかけることを前提に購入して，CM-64内蔵のリバーブを切って，TapDelayを利用し，全体に外づけのリバーブをかけるのも，ひとつの手かもしれません。

CM-64常用エクスクルーシブ一覧

○オールパラメータリセット

```
.roland_exclusive 16,22= {$7f,0,0}
```

トーン，パッチ，システムパラメータが初期化されます。また，ピッチベンドが中心に，ホールドがオフに，モジュレーションがオフに，エクスプレッションが最大になります。

初期化命令ですので，MIDI楽器はCM64_を使うという宣言のつもりで書いてください。

なお，CM-32L（MT-32側）もリセットされます。

○マスターボリューム

LA音源，PCM音源で個々に独立しています。もちろん，CM-32LやCM-32Pでもこの処理はできます。

LA音源側

```
.roland_exclusive 16,22= {$10,00,$16, n}
```

ＰＣＭ音源側

```
.roland_exclusive 16,22= {$52,$00,$10, n}
```

nは0～100の範囲で設定してください。なお，厳密には違うのですが，Z-MUSICにはFM音源にも同等の命令（.FM_MASTER_VOLUME n / nは0～255）がありますので，CM-64＋FMのデータ作成時，最終的な各音源のバランスをとるのに使うとよいでしょう。

○パーシャルリザーブ

```
.roland_exclusive 16,22= {$52,00,04
                    n1,n2,n3,n4,n5,n6}
```

CM-32Pの各パートのパーシャルリザーブを行います。パラメータは必ず6個必要で，省略はできません。n1～n6までの総和は，必ず31以内におさまるようにしてください。

なお，CM-32L側には，.MT32_P_RESERVEという，同等の命令があります。

○リバーブ

```
.roland_exclusive 16,22= {$52,00,01
                    mode,time,lev}
```

CM-32Pのリバーブ設定を行います。

パラメータの値は，

mode：リバーブモード
　0：Room
　1：Hall
　2：Plate
　3：TapDelay

time：リバーブタイム
　長くするほど，余韻が長引きます。
　0～7までで，0が1，7が8に相当します。

lev：リバーブレベル
　大きくするほど，余韻が目立ちます。
　タイムと同様0～7までです。

なお，CM-32L側には，.MT32_REVERBという同等の命令があります。

○パートセットアップ

```
.roland_exclusive16,22= {$52,00,$0a
                    p1,p2,p3,p4,p5,p6}
```

CM-32Pのパートセットアップを行います。

pnの値は，0～16でそれぞれ，CM-32Pのパートを1～16ch，あるいはOFF（16）にするかの設定です。

CM-64（あるいは32P，L）のみを使用するだけなら通常は指定しない命令ですが，別の楽器と並用するときに指定すると便利です。

なお，CM-32L側には，.MT32_PART_SETUPという同等の命令があります。

まとめ

ローランドの楽器について書いてといわれても，ほかの楽器を知らなければそれが標準に見えちゃうんですよね。

私はローランドだけでしか使ったことありませんし，CM-64，いや，特にD-70（こいつはどっちかというと，MIDIキーボードなので，ほかの楽器を制御する側なのよ）のマニュアルには，MIDIについてわざわざ書いてはあっても，ローランドのことが標準のように書いてあるんだもの。ずるいよ絶対，これ見てたから知識が偏ったんだよなあ。

エクスクルーシブも転送は面倒くさいですが，おまじないのようなものだと考えてしまえば意外と単純です（実は自分がいままで敬遠していたという……）。これを知ることが，よいデータを作るコツだとはいいませんが，今後のLIVEの投稿や，BBS上に回っている音楽データで，面白いのが見つかるのを楽しみにしています。

市販アプリケーションを使う

MUSIC PRO-68K［MIDI］と仲よくなろう

Takahashi Tetsushi 高橋 哲史

DTMというイメージにもっとも近いX68000用ソフトMUSIC PRO-68K［MIDI］。ここではこのツールとの賢いつきあい方を見てみましょう。同時にMUSIC PRO-68Kの問題点も探ります。

世には音楽ソフトの多かれど，楽譜入力ツールの，ああ，なんと少ないことよのぉ。ということで，最近Macintoshの贅沢な環境を横目で見ながらも，X68000唯一の楽譜ワープロMUSIC PRO-68Kをせっせと使い込む私なのでした。

以前からいろいろいわれちゃってるMUSIC PRO-68Kですが，使い込んでくると案外かわいいやつなんですよ，ほんと。ただちょっと最初はとっつきにくいところや(やっぱり少々MIDIの知識もいりますしね)，細かい落とし穴や罠があったりして敬遠されがちなだけで……。

てなわけで，MUSIC PRO-68Kを使って23年っ（うそ。本当は3年くらい）のこの私がMIDI特集に便乗して上手なMUSIC PRO-68Kとのつきあい方を講釈させていただこうという趣向なのでございます。

とりあえず，初心者の引っ掛かりがちな罠を見てみましょう。

音が出なけりゃ話にならない

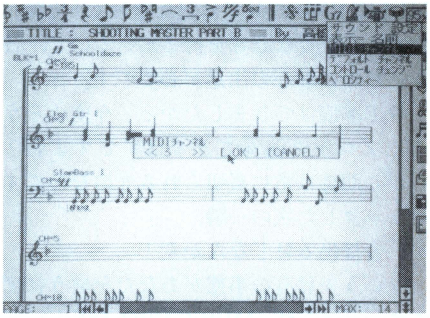
まず図1をご覧ください。これは6月号の付録ディスクについてきたSION IIのMT-32用BGM，B_SHOOTINGMASTER.SCOの1ページ目です。上から順にメロディ，コード，ベース，未使用，ドラムとものすごくわかりやすい構成になってます。

SHOOTINGMASTERはもう完成した曲ですから，このページだけブロック演奏しても正常に演奏されるはずですね。そうですね？ さて本当に自信ありますか？ はい，それでは早速試してみましょう。演奏アイコンの中からブロック演奏を選んでこのページをクリックしてみます（キーボードショートカットのスペースバーを使用するのが「通」かもしれませんが）。すると……，

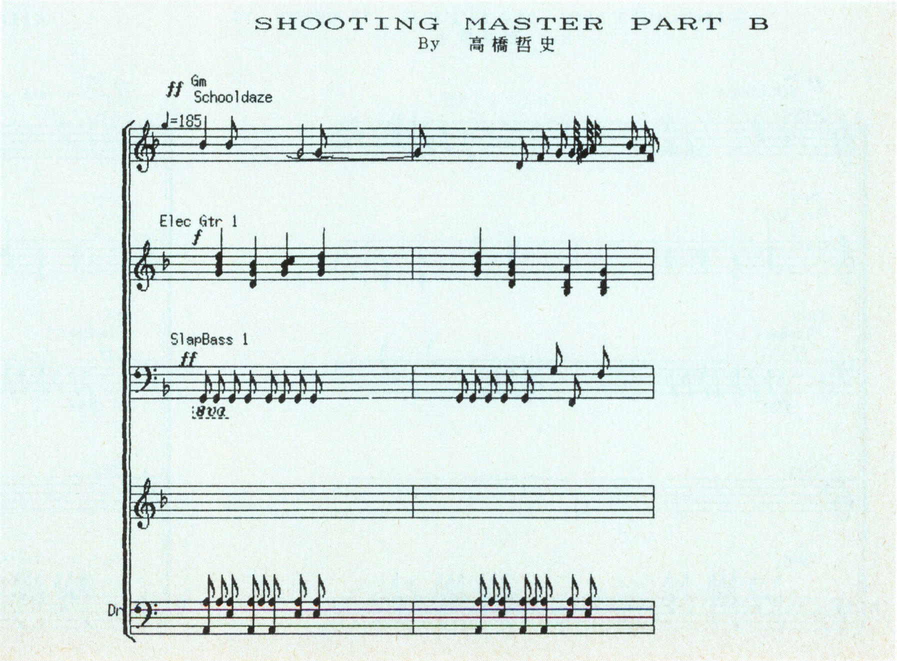
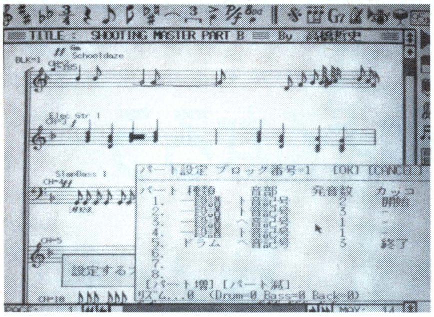
なんにも音が出ないっ！

という不可解な状況にあなたは遭遇することになるでしょう。いくらクリックしてもMT32はうんともすんともいってくれません。えぇーうっそー！ なんで，どうしてどうして？

えー最初からちょっと意地悪してしまいましたが，図1は演奏されなくて当然なのです。なぜってこの私がわざと鳴らないように書き換えたから(笑)。よぉ～く目をこらして見るとこの楽譜にはMIDIチャンネルの設定がないんですね（付録ディスクについてきた楽譜と見比べると一目瞭然です）。受信チャンネルが云々といったMIDIに関する基礎知識はほかの記事にゆずるとして，とりあえずMUSIC PRO-68K［MIDI］ではパートごとにチャンネル設定をしなければ鳴らないのです。

知っている人には当たり前のことかもしれませんが，私は最初それがわからずにハマりました……（だってずーっと内蔵音源版のMUSIC PRO-68Kを使ってたからチャンネル設定なんて知らなくて)。

チャンネルを選択

ということでサウンドアイコン（＝トランペットの絵柄のね）からMIDIチャンネルを選んで，上のパートから順々にCH＝1，CH＝2，CH＝3，……と設定していきます(図2)。あ，細かいことですが，チャンネル設定はそれぞれのパートの頭に書いておかないと正常に演奏されませんのでご注意くだ

図1

パートごとの発音数を設定

さい（図3参照）。

チャンネル設定だけにとどまらずMUSIC PRO-68Kでは記号をおく「位置」が非常に重要な意味を持っているので，書き込む際は細心の注意が必要です。

さて準備はできましたか？　今度こそうまくブロック演奏できるはずですよね？　しかし期待は常に裏切られるものなのです。クリックした次の一瞬，

音は出るけどなんか変！

という未曾有の事態があなたを襲うはずです。実はこれ，MUSIC PRO-68KというよりはMT-32側の問題なのですが…。よく聴いていただくとおわかりになると思いますが，変に聴こえていたのはドラムパートがむちゃくちゃなメロディ（？）に変わっているのと，本来のメロディパートはうんともすんとも鳴ってないというこの2つの現象に起因しているのです（パート演奏を使うとはっきりと確認できます）。

これはどういうことなのかといいますと，MT-32の初期状態（電源ON時）の受信可能チャンネルは2から10で，10チャンネルは必ずリズムパートがこなければならないように設定されているからです。つまり1チャンネルに設定したメロディパートのデータはまったく無視され，10チャンネルに設定されるべきだったドラムパートを5チャンネルに設定したので，むちゃくちゃな演奏をしてくれちゃったのです。

というわけで，上から順にCH＝2，CH＝3，CH＝4，CH＝5，ドラムパートにCH＝10を設定してやっと正常な楽譜（付録ディスクに収録されていた状態），とあいなるわけです。音源によってベーシックチャンネル（受信可能なチャンネル）は違うので，インプリメンテーションチャートできちんと確認しておきましょう。

あとマニュアルにも書いてある（118ページ）ことですが，音が出る出ないでもうひとつ注意しなければならないことがあります。それはパートごとの発音数です（五線譜のアイコンで設定します）。MUSIC PRO-68Kはここで設定された発音数をオーバーして発音することはできませんので，まず最初にしっかりと発音数設定をすることが重要です。基本的なことですがうっかり見逃すとハマってしまいますぞ。

曲にならなきゃ意味がない

さてとりあえず音は出るようになりました。しかしここからがMUSIC PRO-68Kの本番です。「こう鳴るはずなのに思いどおりに鳴らない」，「通常の楽譜での表記はこうだけれどMUSIC PRO-68Kではなんか違う」といった症状が続々と（というとちょっといいすぎか）出てくるからです。そんななかでも，ちょっと引っ掛かりそうなところを，ピックアップしてみたいと思います。

まず注意しなければならないのは音符の数です。MUSIC PRO-68Kでは4分の4拍子とか12分の8拍子などの拍子記号が書けますが，実は通常のシーケンサのように入力時の拍子チェックは行っていないので，1小節中にいくらでも音符が書き込めてしまうのです（しかもきっちり演奏してくれる）。

極端な話をすれば，1小節中に全音符が10個でも書けてしまうのです。さすがにそんな間違いはすぐにわかりますが，やっかいなのは細か～い打ち間違いです。1小節中に32分音符が33個あったなんてのはそうそうわかるもんじゃありません（普通の曲だとそんなに細かい音符は使わないけど，ゲームミュージックの場合は結構あるんですよね）。演奏してみてリズムに「引っ掛かり」を感じる場合はこの手のミスであることがままあります。

次にやっかいなのが先ほども使った「ブロック演奏」です。曲を作っている最中はもっともよく使う機能といってもいいのですが，実はさまざまな制約をしょって立っているんですね，これが。

マニュアルに「ブロック演奏では，そのブロックに入力された記号のみの確認に用いてください」とありますが，本当にブロック演奏は悲しいまでにそのブロックしか

図2

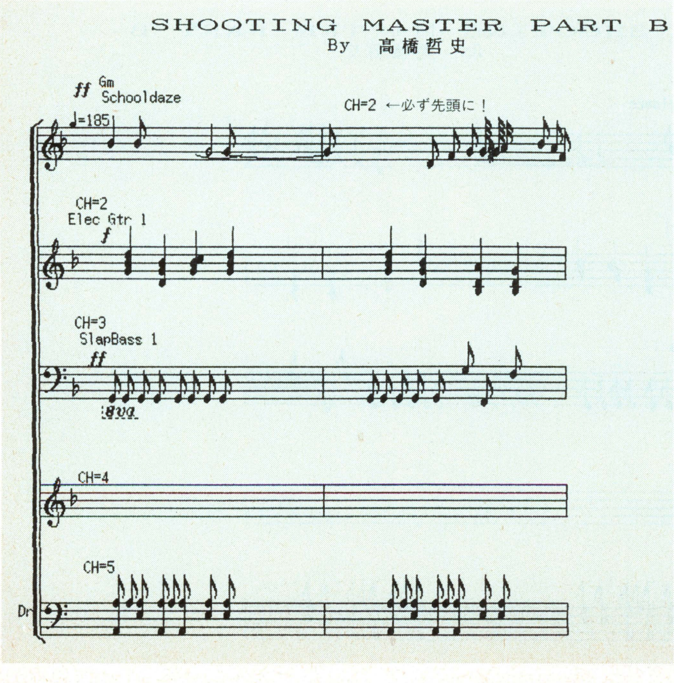
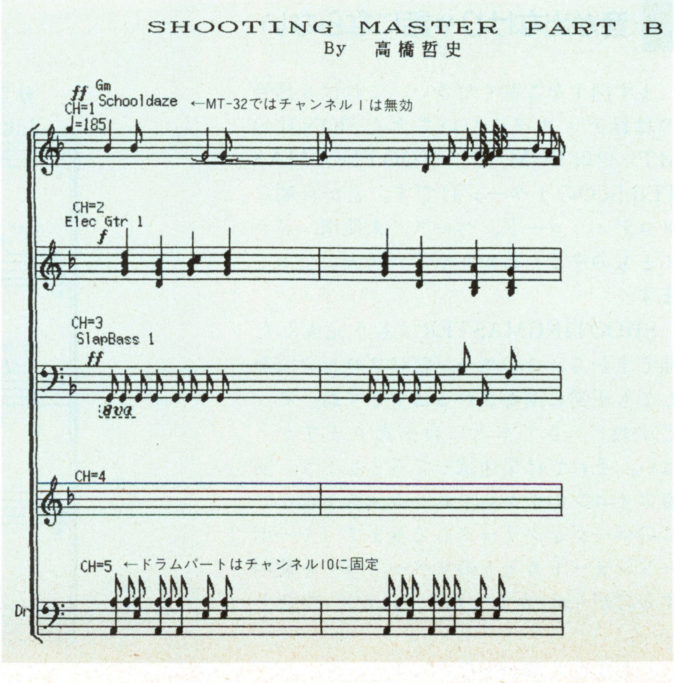


図3



見ていないのです。したがって，テンポも音色もオクターブ記号も発想標語もコントロールチェンジも，そのブロックで設定されていなければ，直前に使用した値を勝手にセットして演奏してしまうのです。曲のイメージを確認するのに，勝手なテンポの勝手な音色でやられてはかなりつらいものがありますよね。

そういうわけでまっとうにブロック演奏するには，「このブロックはこう鳴るはずだ」という設定をすべて人間側でそのブロックに書いてやる必要があるのです。これはかなりつらい妥協ですが，仕様なので本当にしょうがありません。本当ならブロック演奏時にMUSIC PRO-68Kが曲の頭からざーっとプログラムチェンジやテンポなどの情報だけを拾っていって，本来あるべき姿でブロック演奏してくれないといけないんですけどね。このあたりは本当に困ったちゃんなんですよね。

それでは最後に「ええ!?」と思うような事実を２つほど。

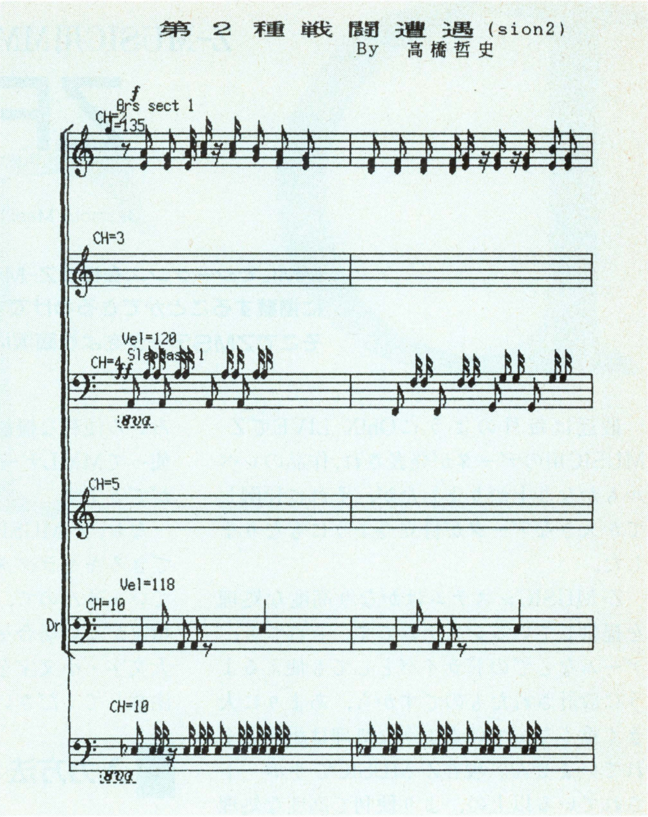
MUSIC PRO-68Kはディレクトリを認識してくれません。それはどーゆーことかというとカレントにあるファイルしか扱えないということなのです。悲しいですがこれも仕様なのでしょうがありません。

それからマニュアルにもありますが，ドラム譜では合計14個のノートナンバーしか送れないので，ドラムに凝りたい場合はつらいです。ただこれには一応の回避法がありまして，ただのパートのチャンネルを10に設定しておくことでどのノートのドラムでも発音できることになります（ONTHE HOP!の楽譜をご覧になるとわかると思いますが，HAND CLAPはこれで出しました）。別にDRUM.DEFを書き換えて鳴らしてもいいのですが，曲ごとにいちいち書き換えるなんて面倒でしょ，やっぱり。

我流MUSIC PRO-68K書き方術

一応前段落までで注意すべきことは書き終えたので，ここでは私が実際曲を入力するときにどういう手順を踏んでいるか，どういうことに気をつけているかを簡単に追っていこうかと思います。３年間の試行錯誤のなかで身につけた書き方ですので，多少なりとも参考になればと思います。

まず私はなにも記されていない楽譜数十ページ分をSCORE.SCOというファイル名で保存しておいて，新たに曲を書く場合はまずそれをロードして書き込んでいくということをしています。どうしてそういうことをするのかといえば，いちいちパート設定やチャンネル指定をブロックごとに書きながら，曲を書き進めるのが苦痛だからです。こうして最初に無記入の楽譜を用意しておくと，1ページだけパート設定を忘れてあとでハマってしまうということがなくなります。

図4

また私は1ブロックに最大2小節ずつしか書かないようにしています。デフォルトは1ブロック4小節なのですが，これだと音符やプログラムチェンジなどがごちゃごちゃ重なって見にくくなることが多いので，SCORE.SCOを用意するときに小節線を消して2小節に直してしまいました。

また，入力時にはキーボードを多用します(47ページダイレクトキー参照)。やはりマウス＋プルダウンだけだとつらいです。左手にマウス，右手にキーボードが基本でしょう（別に逆でもいいんですけど）。

それから流れによってはスポイト機能（38ページ基本操作参照）もよく使います。赤ポインタの正確な位置を記憶するまではちょっとつらいですが，慣れてしまえばこっちのもんです。

それからちょっと変な使い方ですが，私はコードのパートのところによくコードネームを覚え書きしています。本来は自動伴奏用の機能なのですが，No Rhythm状態ならば別に害はないし，ドキュメント性も高くなるので私は好んで使っています。あとで読み返すときになかなか便利ですよ。

そろそろ次のステップを……

さてなんだかんだとMUSIC PRO-68Kについて書いてまいりましたが，やはり現行のMUSIC PRO-68Kではもうかなり手狭になってきているというのが正直な感想です。私のように一度楽譜にしないとどうも曲が書けないという人間にとって，使えるソフトがMUSIC PRO-68Kしかないというのはさびしい現状です。グラフィックだとZ'sのver.3.0やMATIERなんてグーなソフトが出ているのに，楽譜ワープロはいまだにMUSIC PRO-68K止まりで未開拓。なんとかならないものでしょうか。

やっぱり次に望むとすればSX上で動くMUSIC SX-68Kでしょうね。Z-MUSICをベースに，曲の全体を眺めながら（縮小表示で10ブロックも表示してくれればあとはスクロールバーで十分）ルーペを使うようにエディットしたいブロックを選択できて，曲の早送り，パートマスクも自由自在，現在演奏中のポインタも楽譜上ではっきり表示なんてことになるともう嬉しくてしょうがないんですが。

もちろん，一度に複数の曲のエディットもできて，カット&ペーストは複数ブロックを対象にできるんですよね。で，しかもできあがった曲をZMS形式できちんとオプティマイズして書き出せちゃったりすると，これはもーあなた極楽じゃあーりませんか。それと楽譜の清書もちゃんとやってプリントアウトしてくれると嬉しいですねぇ。あ，もちろん同時発音数16以内なんてつまらない制限はなしですよ。できればリアルタイム録音も可能にしてほしいな。

もしこれだけの条件を満たしたバージョンアップ版が出たらメモリをがばっと増設してでも買っちゃうんですけど……。シャープさん，SOUND SX-68Kの次はぜひお願いします。

Z-MUSIC用MMLデータプリプロセッサ

ZPP.X

Okamoto Masakazu 岡本 正和

MMLでシーケンスを行うZ-MUSICシステム。そのおかげでデータを誌面に掲載することができるわけですが，最近のデータはどうも肥大しがちです。そこでZMSデータをより簡潔に記述できるプリプロセッサをお届けします。

最近は毎月のようにOh!X LIVEでZ-MUSIC用のデータが発表され，作品のレベルもかなり上がりましたが，それに比例してか大きなデータが目立つようにもなりました。

Z-MUSICシステムはかなり高度な処理を提供しているシステムです。それでも，ゲームなどでのドライバとしても使えるように設計されたものですから，あまりに大きく重くなっていくような処理は組み込まれていません。現在Z-MUSICでサポートされている以上の，より便利で高度な処理というものも考えられます。そのいくつかはZ-MUSICの拡張によって将来サポートされるでしょう。しかし，Z-MUSICの基本的な性格から，どうしても演奏表現力に関係する部分が中心になってしまいます。またやみくもな拡張は将来データの互換性を保つうえでの障害になってくることも考えられます。

本体の拡張に代わるひとつの方法がプリプロセッサによる拡張です。あくまでもZ-MUSIC本体はいじらず，それに与えるデータをあらかじめ加工しておくことによって機能拡張を行うことができます。

これならば，プリプロセッサ部分を変えていってもZ-MUSICに与えるデータレベルでは完全に互換性を保つこともできます。このようなプリプロセッサによるドライバの拡張を行ったもの，それが今回紹介するZPP.Xです。

このツールは基本的にZMSファイルをより簡潔に記述するためのツールです。Z-MUSICに対するプリプロセッサとして働き，ZPPによる拡張MMLファイル（拡張子.ZPP）をZMSファイル（拡張子.ZMS）に変換します。

大きな特徴としては，ブロック呼び出し，大文字アルファベットのマクロ展開，&コマンドやポルタメントコマンドの@Bコマンドへの展開モードなどです。その他，いろいろ便利な機能もあり，これらの機能を使ってMMLデータをより簡潔にすることができます。

なお，Z-MUSICではMMLの記述に使用できるキャラクタセットはほとんど残されていませんので，このプリプロセッサでは指定がない場合を除いてアルファベットの大文字・小文字を区別します。あらかじめ注意してください。

入力方法

ダンプリストはLHAによって圧縮されています。6月号の付録ディスクなどに収録されていたMAC.Xを使ってリスト1を入力してください。全部打ち込み終わったら，ファイルサイズ11599バイトでセーブしてください。できあがったファイルをLHA.X（6月号のディスクに収録）で展開すればZPP.Xが得られます。

使い方はコマンドラインから，

zpp [switch] <ZPPファイル> [<ZMSファイル>]

です。オプションとして“-d”をつけると出力ファイルに展開情報（元の行番号・ブロック名など）をつけて出力します。

拡張MML

以下のコマンドは必ず行の最初になければなりません。

●展開開始・終了

#ext_on ～ #ext_off に挟まれた範囲を展開します。以下の機能はこのコマンド内でのみ有効です。また，逆にこの範囲内に拡張コマンド以外のコマンドがあった場合，出力しないか，エラーが発生します。

●ブロック登録・終了

ブロック名

**の中にそのブロック名を登録します。MML中では[ブロック名]で呼び出されます。そのため，なるべく制御コマンド（[d.s.]など）と重ならないようにしてください。重なった場合はブロック名として扱います。

**

ブロックの終了宣言です。なお，宣言がなくても続けてブロック登録宣言を行えばブロック終了と見なします。

ブロック名登録宣言から終了宣言までのMMLデータをブロックとして登録します。特にチェックはしていませんが，ブロック名に全角文字を使うのはあまりおすすめできません（内部ではすべて1バイト文字として処理するため）。また，当然ですが絶対に] を使ってはいけません。

( )コマンド

( )で括ってあるコマンドです。行中にいくらあっても結構です。( )で囲っていない文字があれば，そこからMMLデータと見なします。解読できないコマンドがあるときは無視して出力もしません。また，マクロセットコマンド以外，大文字，小文字の区別はしません。

●トラックセット

(Tn)

トラック番号をセットします。このコマ

図1 ブロックの例

```
*a1*    MMLデータ1          /ここから[a1]ブロックです。
           :
           :                /ここまで[a1]です。
*a2*    MMLデータ2          /ここから[a2]ブロックです。
           :
           :                /ここまで[a2]です。
**                          /ここは[a2]には含まれません。
```

ンド以降の文字列はMMLデータと見なします。なお，以前セットしたトラック番号と異なる場合，新しいトラックが始まったものとして，マクロ以外のパラメータを初期化します。したがって，このプリプロセッサでは一小節ごとにトラックを入力していくような入力形式は向いていません。

●デバッグコマンド補助

(Dn1［,n2］)

［!］コマンドの出力を制御します。トラックの最初からn1，n2回目に現れた［!］コマンドのみ出力し，ほかは出力しません。n1のみ指定した場合はn1回目とその次を有効とします。n1に0を指定した場合，［!］コマンドをすべて削除します。

●ポルタメントモード指定

(@Pn)

ポルタメント展開モードを指定します。0ならそのまま，1なら@Bコマンドに展開します。なお，2はGS音源用モードとして今後対応する予定です。MML中でも［Pn］コマンドで同様に切り替えができます。普通はこちらを使ったほうがいいでしょう。それから，このモードはトラックセットによる初期化が行われませんので注意してください。

●キースケール（音調）登録

(@Kn,［音階……］)

Cメジャー（ハ長調）を基準とした音調を登録します。n には0～9が指定できます。［音階……］のところにはフラットやシャープをつけたい音階を並べてください。

例)

(@K0,c+d+f+g+)

Cメジャーを基準としてC，D，F，Gを半音上げる（要するにEメジャー）。

●大文字アルファベットマクロ登録

(#［A～Z］,”MMLデータ”)

大文字のアルファベットにMMLを登録します。#の次の文字は必ず英大文字にしてください。なお，マクロ展開はブロック展開の次に行うのでマクロからブロック呼び出しを行うことはできません。また，マクロの中にマクロを含むこともできません（マクロ内の英大文字は小文字に変換します）。マクロを登録していない場合，初期値としてそのマクロを小文字にしたものを持っています(Aには“a”が登録してある)。

特殊MMLコマンド

通常のMMLのほかに以下に説明するコマンドが使用できます。

●ブロックコール

［ブロックネーム］

ブロックを呼び出します。呼び出したブロックからさらに違うブロックを呼び出すこともできます。ただし，再帰呼び出しを行おうとするとエラーとなります。もしそのブロックネームが登録されていない場合，Z-MUSICのMMLとしてそのまま出力します。

●特殊展開

”n11［,n12］［n21,n22,……］MMLデータ”［”……”］……

ZPPの特徴のひとつです。ZPPでは同じトラックから何回そのブロックを呼び出したかをカウントしています（トラックが変更されると初期化される)。このコマンドは，そのブロックの呼び出し回数と，呼び出されたトラック番号によって展開する内容を選ぶためのコマンドです。”に続けて，呼び出し回数，トラック番号，……と数字を並べます。条件が一致したら””に記述してある内容を展開します。続けて条件を並べるときは，”1,12 .....””2,12 .....”と，”のあとに空白を入れずに次の条件を入れてください。ZPPは前から順番に条件を比較していき，最初に条件があったものだけを展開します。このコマンドはこのバージョンでは2行以上に分けて記述することはできませんので注意してください。また，このコマンドはブロック内でのみ意味を持ちます（使ってもエラーではないが，使う意味がない。ちなみにその場合呼び出し回数は不定）。

●英大文字マクロ展開

A～Z の英大文字

マクロ登録で設定したマクロを呼び出します。ただし，A～Zの文字の前に@があった場合，その文字はマクロと見なしません（そのまま出力する)。注意してください。また，展開順位の都合上，マクロの中にブロック呼び出し，特殊展開のコマンドがあっても無視されます（たいていZ-MUSICでエラーが出る）のでそれらのコマンドを含まないようにしてください。

●キースケール（音調）セット

ks［基準音階］［登録番号］［m］

登録したキースケールを呼び出します。基準音階を省略するとCとなります。登録番号を省略するとメジャースケールがセットされます。ただし最後にmをつけるとマイナースケールとなります。mをつけたときは登録番号はつけないでください。パラメータを省略するとCメジャーになります。

例)

(t1) ksfm cdefgab<c

/キースケールFマイナーを指定

［実際の出力］

(t1)　　cd-e-fga-b-<c

●ナチュラル

［音階］%

音階に続けて%をつけることによって，強制的にCメジャーの音階となります。通常ナチュラルとして使用しますが，%に続けてフラットやシャープを指定してもかまいません。そのためここでは相対音長指定として2進数を使うことはできません（そんな人はいないと思いますが)。

例)

(t1) kscm cdefgab b%

図2　特殊展開

```
例)
    (t1) [a] [a] [a]
    (t2) [a] [a]

*a* "3,  1  c"",  2, 1 d"" e"
**/  |   |   |    |    |パラメータを指定しなければ無条件に展開する。以降
     |   |   |    |    └は続けても無駄。
     |   |   |    |
     |   |   |    └展開する条件を複数つけたいときは続けてパラメータを記述する。
     |   |   |     また，パラメータを省略または0としたときは常に条件成立と見なす。
     |   |   |
     |   |   └続けて条件をつけるときは空白を入れずに続けて次の分を記述する。
     |   |
     |   └展開するトラック番号を記述する。
     |
     └ブロック呼び出し回数を記述する。

  この例では以下のように展開されます。
(t1)   d
(t1)   e
(t1)   c
(t2)   d
(t2)   d
```

ZPPの展開について

ここではZPPが実際にはどのような展開を行っているかを説明します。通常のCコンパイラなどのプリプロセッサとはかなり違う（当然ですが）ので，よくZPPの動作を理解してください。

●全体の動作

ZPPは2パスで展開します。1パス目ではブロックを登録します。ブロックとして登録した部分は2パス目ではブロック呼び出しをしない限り参照されません。また，ここではマクロ登録，展開は行いません。

2パス目で実際の展開を行います。#ext_on/#ext_offで囲まれた範囲以外はそのまま出力します。先頭から順に処理していきますが，マクロの展開だけは注意してください。例を挙げます。

```
(#A,"cde")(t1)  [a]
(#A,"fga")(t2)  [a]
*a* A
**
```

これはこう展開されます。

```
(t1)    cde
(t2)    fga
```

マクロは実際に呼び出された時点で展開されるため，このようなことになります。内部処理は，( )コマンド，特殊展開，[Pn]コマンド，ブロック呼び出し，マクロ展開，キースケール変更，ポルタメント展開の順に行っています。そのため，レベルの低い処理がレベルの高い処理に影響されることはありません。

●&，ポルタメント展開について

ZPPは常に@K，@Bコマンドに注目して，その値を記憶しています（トラックが変われば0に初期化される）。ポルタメントモードが0のときは特にそのほかの動作はせず，そのまま出力します。モードが1のとき，&・ポルタメントを@Bコマンドに展開します。基本的にピッチベンダの値を考慮して展開しますので，&・ポルタメントを伴ったデチェーンもFM音源と同じ感覚で行えます。ただしNコマンドなどでチャンネルを変えた場合，ベンド値が変わっている場合があるとうまく動作しませんので注意してください。

また，モード1のとき，&の次に@B・@Kコマンドを入れるのはなるべく避けてください（それなりの処理はしていますが，保証はできません）。

次のようにMMLが与えられた場合，次のように展開します。

```
(t1) [p1]  @b0 c&d&(de)48&(ed), 20 c
        ⋮
(t1)    @b0 c& @b683c& @b683, 1366c *48& @b1366, 683, 20c @b0c
```

最後に

もともとのZMUSIC.Xはマクロ機能などは持っていません。本来，そのような役目はMUSICZ.FNCが果たすことになっていたのです。BASICであれば，単なるマクロ展開のみならず，変数の使用，文字列処理などあらゆることが可能だからです。しかし，処理プログラムの記述のためにプログラムが肥大化することも考えられます。ZPPのように機能を限定することでプリプロセッサを独立させたほうがデータ量は確実に縮小されます。

さらに最近はZMSファイルによる投稿の比率が非常に大きくなっているようです。これは終始慣れたエディタ上で作業できることのメリットによるものと思われます。誌面に掲載する場合でもZMSのほうがデータが簡潔になります。ZPPによって，さらに簡潔な表記が可能になりますが，半面，データの可読性が低くなるというデメリットもあります。OPMDRV時代にはX68000のデータを他機種に移植することは不可能ではありませんでしたが，Z-MUSIC用のデータは可搬性が低くなっています。それに追い打ちをかけることにもなりそうです。

今後はOh!X LIVEでもZPPを使用した投稿も受け付けます。長大なデータで困っている人はZPPを活用してみてください。

リスト1 ZPP.LZH

```
0000  1E 7C 2D 6C 68 35 2D 2E  : 2B
0008  2D 00 00 8E 53 00 00 00  : 0E
0010  60 6D 18 20 01 05 7A 70  : F5
0018  70 2E 78 93 F7 48 00 00  : E8
0020  24 97 84 DE BE FB AC 69  : EB
0028  49 3F DF AE E5 CB 3C 59  : 5A
0030  64 7C 59 64 6D 99 18 79  : 34
0038  B2 C8 C6 10 97 68 88 84  : 5B
0040  D9 89 77 E9 01 B2 E4 C9  : 22
0048  23 76 27 4D 17 17 D0 C9  : D4
0050  3D E1 01 92 F4 80 22 6C  : B3
0058  DC 7D 7E B0 18 00 00 00  : 9F
0060  B8 6B 4F 66 BA EB 5D 8E  : 68
0068  8E EA C4 C1 15 78 C3 A4  : F1
0070  30 4E 31 E5 CD 13 1A 15  : A3
0078  77 30 4C 31 00 4D 74 4C  : 31
-------------------------------------
SUM:  A0 61 EC 62 1A 55 B3 EE  9561

0080  35 D3 94 D8 4F 7F BF FF  : 00
0088  BD 77 20 01 8B 8E BB BB  : E4
0090  95 D5 DB 55 BB 5D DB F0  : 7D
0098  77 7C 1B D5 5D DF 1F 55  : 93
00A0  75 57 7F 01 FB E8 EF 9B  : B9
00A8  BE 7E FE 36 67 77 69 64  : 1B
00B0  72 36 DC 5E E0 D3 3F 51  : 25
00B8  F9 46 35 65 11 8D 39 E4  : 94
00C0  98 D5 54 26 33 8D 59 BF  : BF
00C8  C7 F3 1E 04 99 34 FA 39  : DC
00D0  50 D8 F0 31 BD F6 86 47  : C9
00D8  7B 55 2F 45 2F 52 3E EB  : EE
00E0  25 E6 65 E6 A5 E5 F3 16  : E9
00E8  99 FE C6 9E 14 23 60 EA  : 7C
00F0  AD A4 F8 19 FE F5 6C DE  : 9F
00F8  81 2E CA 0F 63 D9 25 D8  : C1
-------------------------------------
SUM:  B2 97 B6 49 17 E7 3F 13  DB02

0100  E4 B2 58 E4 B3 5E 02 5F  : 44
0108  5A 9E BF 3B 65 DF 5B 68  : F9
0110  2D 9D 1B 5A B7 DA 93 E2  : 45
0118  43 A8 CC C9 92 97 73 2E  : 4A
0120  C6 5D 50 CF 29 F5 D2 EE  : 20
0128  76 57 BF 6B D7 97 A0 97  : 9C
0130  0C 58 6A BC 4D C8 B0 D7  : 26
0138  5B 74 AC 0D 83 83 67 83  : 78
0140  E8 D9 B1 F8 18 E1 B1 BD  : D1
0148  63 D6 D6 5B EC FD 9D 66  : 56
0150  9C 79 EE D8 F3 58 F8 0C  : 2A
0158  EB 07 B2 BD E9 E6 98 F5  : BD
0160  B5 2B 6C 7F 83 5B FB 4C  : F0
0168  72 F5 BD FE C7 90 C7 9E  : DE
0170  C6 F8 62 39 2C 7E 16 3F  : 58
0178  D6 29 DC B1 C9 CC D8 48  : 41
-------------------------------------
SUM:  E6 85 B1 94 50 D6 7A 4B  D79E

0180  95 13 33 F5 A4 00 67 A0  : 7B
0188  00 18 3E A6 77 A9 95 E9  : 9A
0190  EA 7C 00 DA 4D 00 7C F3  : FC
0198  60 D8 FB C3 46 7A CC 72  : F4
01A0  FA 80 76 83 BA CB 7D 7E  : F3
01A8  CD 8D AE C3 EF 6A 3D E6  : 47
01B0  3F 89 8F C6 C7 A0 68 2E  : 1A
01B8  8D 2E 54 B6 25 EA 45 AF  : C8
01C0  96 C7 20 59 FA 4C 7E 73  : 0D
01C8  59 1B A2 27 BF D0 7B C6  : 0D
01D0  3F 8D 8D D3 1F 75 8F 20  : 6F
01D8  D9 F3 66 6C 2C 0C 6E FC  : 40
01E0  C0 6C 6A EC C3 37 D1 78  : C5
01E8  F4 2F A9 69 CC 85 25 A9  : 54
01F0  9D BE A9 33 6D 03 6D BE  : D2
01F8  99 5A B8 BA 4B E8 BD B7  : 0C
-------------------------------------
SUM:  63 58 9C FB 8E 26 C1 1A  3E9B

0200  CA 63 63 2A 2B B6 43 CF  : AD
0208  34 7A 7C 13 D8 E6 4F E7  : 31
0210  E5 9E 9F 58 A9 67 04 7A  : 08
0218  62 0A 74 66 6D 5D E9 0C  : 05
0220  1E C6 BF E2 32 49 67 8C  : F3
0228  E1 14 6E AA 4D 8D 59 01  : 41
0230  4C 15 4C 6D D0 10 B7 2B  : DC
0238  78 F1 51 46 2A F0 A1 C9  : 84
0240  DC EF EE 74 10 2E 98 CD  : D0
0248  18 DE 46 FD 7C 7F EC 20  : 40
0250  E6 BB FA 92 6E 6D 69 BC  : 2D
0258  43 FF 8D 32 BF F3 FF 7F  : 31
0260  9F 8B A4 E7 AF 81 B1 F9  : 8F
0268  FE 29 E9 F4 30 36 3F 3F  : E8
0270  F3 1F 63 B8 3D 39 33 D3  : A9
0278  E0 7E 8D 8F 87 BD 33 70  : 61
-------------------------------------
SUM:  95 3D F4 91 EE F0 D9 60  57CB

0280  5F 2B 22 E0 B5 8F 9D EA  : 57
0288  2B 3E 30 BE 29 EF F3 31  : 93
0290  74 6C B2 D6 26 C2 BC 5A  : 66
0298  FF 01 60 E2 DD F1 57 A3  : 0A
02A0  C7 A0 68 33 91 7C 03 1A  : 2C
02A8  BF 2A 7B FB DB 0A F8 CB  : 07
02B0  AB 42 30 72 A0 CE 51 D4  : 22
02B8  8B 78 D0 67 0A E6 9A 9D  : 61
02C0  29 C0 25 76 A3 DE A5 BD  : 67
02C8  6A 53 96 15 F0 4C 7F 3B  : 5E
02D0  9C 55 D9 8D 22 35 85 7A  : AD
02D8  AF 8E 13 8C 56 B4 FD DF  : C2
02E0  44 7B 1D C3 43 BF 4D 71  : 5F
02E8  83 D2 5F 62 78 AB C1 C7  : C1
02F0  A1 7B 81 8D A6 A7 52 84  : 4D
02F8  6D 76 66 EA 11 7A 6E 80  : AC
-------------------------------------
SUM:  6C 8E 51 9D 74 09 FD FB  7262

0300  2F 76 5A 0C 70 7F 22 6E  : 8A
0308  83 F9 B5 D8 DE BC FF 5A  : FC
0310  0C E7 FD B5 AD 77 FC 35  : FA
0318  A9 57 04 D1 39 DC F8 BD  : 9F
0320  6D 8B 37 DD 6D 83 2C A7  : CF
0328  38 5D 3B 22 3B D9 CF 66  : 3B
0330  A4 D6 5D 6A 60 A8 6D 59  : 0F
0338  BE 3E 1E A5 E4 E4 59 51  : 31
0340  6F 38 B2 BA F5 DF 1A C6  : C7
0348  BC B4 8D EF 3F E7 E8 78  : 72
0350  F3 9B 8A 93 75 96 DB AC  : 3D
0358  B7 CA 5B 24 81 B5 94 4A  : 14
0360  59 C7 90 EE 46 0B 7F 6C  : DA
0368  59 3F 3F E8 EB BB DB A8  : E8
0370  2E CE 25 D1 A5 0D 38 5B  : 37
0378  E3 4C F5 A9 C7 61 C8 63  : 20
-------------------------------------
SUM:  06 1A 0A 28 E7 BB A1 77  1D37

0380  10 B4 9F A7 EC 5E 24 B9  : 31
0388  6F 53 8E 65 FC E0 58 39  : 22
0390  22 EA 58 BA 37 13 73 BF  : 9A
0398  6C B1 A3 8E D8 8F E0 16  : AB
03A0  85 2B 37 7A 92 A5 1E 31  : E7
```

```
03A8  53 3B 9B AD E5 3E C8 B3  : 74
03B0  C9 AE EE EB B4 38 BB 74  : 6B
03B8  B1 7D 9D 9A 55 09 62 ED  : 12
03C0  DD 18 D0 AD 8C E6 0C 13  : 03
03C8  21 0A 2C 5C D9 9B 67 E7  : 75
03D0  F8 1E 91 06 52 89 05 10  : 9D
03D8  6D 3C 13 8B EA 1B E4 9A  : CA
03E0  29 70 23 46 BB 43 06 72  : 78
03E8  36 48 BD 56 76 3B 94 CD  : A3
03F0  69 1B 1A F9 99 9A E2 CC  : 78
03F8  F4 FA 9C FA 7F 88 D7 AC  : 0E
------------------------------------------
SUM:  7E 7C BB 29 61 C9 81 67   8E2B

0400  2A 2C 0D 8A 40 DA 3C 4D  : 90
0408  76 85 28 23 47 19 D1 A2  : 19
0410  F1 CC F9 28 26 76 7E 47  : 3F
0418  5A 36 1D 6A B2 E5 79 15  : 3C
0420  D0 5E B5 DC B0 A9 AC 2B  : EF
0428  DD E1 CC DD 5A 08 45 B8  : C6
0430  D2 2D 5E 35 1E EE 0B D3  : 7C
0438  DE A8 5C 02 7E 17 AE 16  : 3D
0440  BB 69 B9 1A DF 79 F1 AB  : EB
0448  B4 34 9A 0C 5D 9F 61 83  : 6E
0450  C5 DB 9A 6F 83 43 40 89  : 38
0458  73 01 BA 2F 92 08 97 BA  : 48
0460  DC 97 62 AA 81 6B DB 7B  : C1
0468  95 1C B5 B1 5D CC DF 6B  : 8A
0470  50 C1 36 38 86 F4 70 70  : D9
0478  60 3F 75 33 67 E9 4B 62  : 44
------------------------------------------
SUM:  10 F3 EF B9 21 7B 4C 40   3798

0480  A9 00 B4 8D BE 9C B4 8F  : 87
0488  4B 0B B2 E8 61 5E DF ED  : 7B
0490  74 EF F6 BC E7 EA DA F7  : B7
0498  DC 9F A3 6B DF 7B AD 64  : F4
04A0  47 66 A0 CF C6 5D 9A 12  : EB
04A8  B3 7B 17 8E FA 86 90 6C  : 4F
04B0  1A 50 95 98 C4 03 80 AB  : 89
04B8  AA 8D CB 2C C9 6C A1 9B  : 9F
04C0  51 FD 5B 83 43 66 F8 52  : 1F
04C8  BB 37 B9 6E 53 37 DD B7  : 37
04D0  29 9B D8 58 7B A3 E1 F3  : E6
04D8  D4 E4 74 24 4E 7D FD CF  : E7
04E0  B1 76 95 84 14 60 B8 38  : A4
04E8  87 70 73 89 71 24 CA 3C  : 8E
04F0  8C BD B2 18 E7 BF 8A 40  : 83
04F8  06 B9 C7 3D D9 48 92 CC  : 42
------------------------------------------
SUM:  D5 66 F7 8C D6 F9 B6 E6   1855

0500  63 2D 98 85 4E CD F7 12  : D1
0508  1F D5 01 95 76 DF E6 15  : DA
0510  D1 74 02 BA 29 81 5D 13  : 1B
0518  31 A7 15 42 A3 CC 66 F4  : F8
0520  0C E4 BB B5 33 66 6F 55  : BD
0528  EE F7 15 2C B2 AD 88 0D  : 1A
0530  DF 52 56 5F 87 69 BF B4  : 49
0538  3B FA B5 8D D0 03 AC EA  : E0
0540  31 95 76 9F 70 30 28 8A  : 2D
0548  59 51 D1 0B 51 9C 24 52  : E9
0550  D6 F2 B2 2F 99 E4 17 27  : 64
0558  91 06 D2 5B 19 1D 01 B4  : AF
0560  4D A8 F2 E3 AB 5E A3 87  : FD
0568  0A 53 49 67 22 9C 1A AE  : 93
0570  63 D6 46 C3 AB F0 9E CF  : 4A
0578  18 B4 02 53 9A A9 68 99  : 65
------------------------------------------
SUM:  5B A7 D9 77 51 D8 29 82   4704

0580  52 81 E6 1F 6D 52 6C 95  : 98
0588  AD 02 4D F8 AB EF D8 DF  : 45
0590  AF 16 52 50 E2 72 D1 DF  : 6B
0598  C0 42 3D 60 ED D8 36 BC  : 56
05A0  20 8D 16 E7 D4 FB FB 0A  : 7E
05A8  F8 06 A4 D5 02 99 40 85  : D7
05B0  F8 9A 2A 08 26 C6 4F C1  : C0
05B8  34 1A 28 26 EC 91 9D 8A  : 40
05C0  BE 42 18 35 9C 14 5C E9  : 42
05C8  8F 8F EC 5E 58 57 A5 0F  : CB
05D0  D8 BD 04 C3 A3 F6 85 35  : AF
05D8  90 B8 38 74 65 BE 9C A1  : 54
05E0  71 3F 3A 74 40 B9 B5 CC  : D8
05E8  A3 82 2D 8A FC DC 79 5B  : 88
05F0  15 FE B0 E1 9B 74 3F 25  : 17
05F8  E5 67 CA 59 B2 DB F9 16  : 0B
------------------------------------------
SUM:  75 8E EF B3 54 79 FA 19   9D95

0600  81 F9 E0 0F F0 EC 2B E8  : 58
0608  E1 FB 47 05 90 A7 69 77  : 3F
0610  A0 C1 41 6E 2E 5C 95 CA  : F9
0618  50 D1 71 BF 85 C8 34 F8  : CA
0620  47 AB 41 9B 6B B7 CE 67  : 25
0628  E7 64 4E 67 73 0D 08 6E  : F6
0630  30 5F F7 11 C1 72 2D F0  : E7
0638  53 44 3A F9 82 19 13 99  : 11
0640  CD ED 6F BD A3 F9 2B B4  : 61
0648  3B 2F 35 8D C4 BD 08 66  : 1B
0650  A7 35 2E B8 31 0F B5 63  : 1A
0658  C7 3B 0C DE 81 A6 DA 6C  : 59
0660  74 EC 7A 81 8D EF 20 31  : 28
0668  69 F8 26 D9 EB B2 DF 6B  : 47
0670  D2 D7 FC 1A BD D7 E9 C7  : 03
0678  13 22 72 BC 46 3D 4D 4F  : 82
------------------------------------------
SUM:  3B A1 85 5D E8 26 6A 1A   6C64

0680  84 C5 E0 62 47 C8 98 C5  : F7
0688  94 DB E9 76 C2 90 C5 C3  : A8
0690  1B 5B 9F 84 32 61 F8 C3  : E7
0698  16 DF 7A 63 14 26 A1 2D  : DA
06A0  A3 58 96 3B 11 16 F0 EA  : CD
06A8  1D F4 D2 6D C5 5E 92 DD  : E2
06B0  03 47 AD 76 F4 D4 AA D0  : AF
06B8  03 D5 CE C1 3A EC E7 72  : E6
06C0  10 B1 04 51 39 7F BA 81  : 09
06C8  81 B3 D0 0E 63 A5 69 31  : B4
06D0  B8 EE 35 1C 3B 0A FD C0  : F9
06D8  E6 52 4D FB 39 CC 79 CF  : CD
06E0  5E 0B A6 B2 AA 18 DC A2  : 01
06E8  2B DA C7 F1 EC 1D D8 36  : D4
06F0  31 C2 76 9D 3C 1B 49 B3  : 59
06F8  B0 AF 4F EA 35 A3 E1 62  : B3
------------------------------------------
SUM:  A8 3C 4D 3E 6A 00 80 AF   D845

0700  31 C6 8B 78 D2 86 F8 B2  : FC
0708  BA ED 53 E3 63 C9 30 6F  : A8
0710  E2 EE DF B7 E0 4A E0 30  : A0
0718  D2 8D FF 6F 56 14 6F EB  : 91
0720  42 5C 95 8E D6 89 73 CD  : 60
0728  84 34 4D D1 68 90 9C FB  : 65
0730  ED A6 3F B8 BF D0 AA 15  : D8
0738  AD B3 88 E3 16 7D 1B 02  : 7B
0740  3D 86 39 1D B0 F5 B6 A2  : 16
0748  68 D6 EA B9 00 F4 5B 16  : 46
0750  E1 EB BC E4 0C 81 A4 44  : E1
0758  2B DB 5F 84 8F E4 AD 7E  : 87
0760  8D 02 47 23 C0 DC 96 89  : B4
0768  4A 50 63 35 B0 0B 85 3C  : AE
0770  E4 4C CF 30 8D 6F FB BA  : E0
0778  E8 BA 74 E9 8B 83 84 46  : D7
------------------------------------------
SUM:  53 91 90 2A 51 3A 47 5A   B7AE

0780  9C D8 4A A2 D1 F9 B8 D8  : BA
0788  65 8C 6D 98 3B 38 F8 E6  : 47
0790  85 37 89 36 DF AF BF 95  : 5D
0798  C5 5D A3 0D FD BD 8C 22  : 3A
07A0  B7 4B 1E 69 09 C2 61 3C  : F1
07A8  34 D4 0A 85 9E 84 0A C7  : 8A
07B0  8A ED 11 D5 F9 86 B0 7D  : 09
07B8  1B BC BE BD 40 D3 AB 3A  : 4A
07C0  0E D9 AE 59 91 17 36 D2  : 9E
07C8  A4 67 72 1F DA 27 C7 E1  : 45
07D0  CE 21 20 4E 6D 44 06 91  : A5
07D8  71 64 6E 7E FE 8E D5 9B  : BD
07E0  D2 DA B8 AB 4D 1E 83 BC  : B9
07E8  B6 B5 33 55 AA AC D5 2A  : 48
07F0  99 F1 F9 5D 21 CA E5 99  : 49
07F8  A2 DB 58 5B 64 5A 19 9D  : A4
------------------------------------------
SUM:  8F E0 C4 F9 1A 3A EF 2A   47C2

0800  3F FE 99 CC BC 2A FE 16  : 9C
0808  1E 6B 24 A7 23 BB E6 97  : AF
0810  92 B3 B8 F2 26 65 E4 5A  : B8
0818  FB 3F 30 24 C6 F1 7A 80  : 3F
0820  B4 00 2F EC FD 20 B0 DB  : 77
0828  28 5B AD 91 AF BE DD 8D  : 98
0830  7E 00 56 DA 2C CD 03 0F  : B9
0838  FB 34 0D 27 47 40 A9 40  : D3
0840  D6 15 2D 02 06 3F 2C BA  : 45
0848  25 FE 80 9C 6E 88 4E D4  : 57
0850  CC E8 A6 0A F1 C7 62 25  : A3
0858  F8 55 49 99 F9 0B 89 D2  : 8E
0860  0B 4F 33 2D 15 3C 4E 46  : 9F
0868  12 F5 3B 08 9C 86 7E 3E  : 28
0870  DF 7C CF C9 16 ED 32 C5  : ED
0878  D3 02 31 70 10 AE E6 65  : 7F
------------------------------------------
SUM:  CD FC EE B6 1F 1C C4 71   9003

0880  6D C5 64 B8 B7 05 7B 6E  : F3
0888  B9 E9 3F 42 DB 99 AB 9C  : DE
0890  FD F0 D0 67 D0 F3 58 89  : C8
0898  43 70 5B C4 13 4F EE 03  : 25
08A0  E6 D6 6C 79 1A A5 02 0E  : 70
08A8  3C B6 3E 0F 0F 71 AD F1  : 5D
08B0  47 8F AF D0 26 FF 0F 71  : FA
08B8  E2 43 97 DB 4B D4 4A EE  : EE
08C0  45 C0 73 97 09 47 6C 64  : 2F
08C8  14 9E FA 13 DA 74 ED C3  : BD
08D0  B6 72 29 CC DB 17 33 F9  : 3B
08D8  00 F0 EE B8 15 26 FF 67  : 37
08E0  D3 8F 09 EF A6 04 0D A9  : BA
08E8  FE EB D7 EA E7 91 6C 49  : D7
08F0  4F 39 57 C7 E2 AF 1E B3  : 08
08F8  AE D5 20 6B F5 36 BF 59  : 51
------------------------------------------
SUM:  8E B4 99 91 46 3B 55 79   217E

0900  D8 A7 B0 5B A3 91 A8 E9  : 4F
0908  E4 52 92 81 93 C3 FA 1B  : B4
0910  E5 A8 4E E7 CC 3B 62 C8  : F3
0918  E6 6D 5C EF C0 BD 46 E8  : 49
0920  A2 8D DF A1 BE 57 DF 4B  : EE
0928  09 1D 75 09 10 78 7C 73  : 1B
0930  48 5F C1 B4 5D 29 61 02  : 05
0938  68 FB CA ED 0E A3 D2 D4  : 71
0940  D6 31 B5 D9 5B 6D 3B B3  : 4B
0948  F1 3D F9 C4 93 0A 74 54  : 50
0950  30 2D 4F C2 2F 3A 5B 0E  : 40
0958  34 67 81 5F E5 67 06 D8  : A5
0960  AC F2 56 D4 40 3F E8 CD  : FC
0968  0F E3 63 03 15 38 8E 00  : 33
0970  31 F0 91 7A 47 CF 3D BA  : 39
0978  93 6F E4 99 39 C7 8E 09  : 16
------------------------------------------
SUM:  8C 48 77 A5 D2 0C 29 C5   DD7C

0980  B5 58 0D B8 EC 6C E5 76  : 85
0988  C7 52 0F B5 BB 8A B9 23  : FE
0990  66 00 EC DA CC E1 29 1E  : 20
0998  14 34 9B 71 D7 D7 CA 7E  : 4A
09A0  9F E2 BB C3 5D 6D 0D E8  : BE
09A8  CD 54 4F 12 66 89 A6 BF  : D6
09B0  E1 B3 E8 98 CF 02 09 77  : 65
09B8  8D 38 D7 4C 44 5A 86 8C  : 98
09C0  EA A3 A5 6D E2 AE 7C 7F  : 2A
09C8  60 AB 90 C5 79 8D E3 A7  : F0
09D0  39 F4 EC B7 8F 0F DB AE  : F7
09D8  85 3D B5 AB 81 DA A9 0F  : 35
09E0  15 2D AD 45 2B 5A E6 C7  : 66
09E8  BF EF 47 8C ED E0 5F 3B  : E8
09F0  89 13 8A 7C 67 00 E3 0E  : FA
09F8  01 C5 2A E2 3C E2 9C 42  : CE
------------------------------------------
SUM:  36 72 EA 34 46 40 7A 14   E088

0A00  E9 7C 8C 36 96 13 05 C0  : 95
0A08  57 4B 83 F0 1F 86 5B FC  : 11
0A10  95 E3 47 95 F6 BD 4F 6A  : C0
0A18  17 A2 63 DD CE F4 EE FE  : A7
0A20  D0 B7 91 19 43 33 98 49  : 88
0A28  D0 49 3E 8D 9E 1F 45 0A  : F0
0A30  3A 0B 69 29 67 AD 34 56  : 75
0A38  99 EE 92 16 D9 E2 06 0C  : FC
0A40  E8 7E C9 A2 72 99 E6 D5  : 97
0A48  73 99 FC 68 FE E8 7B 8F  : 60
0A50  DC EA 7F D8 26 E7 B2 BD  : 99
0A58  65 92 EF 3D 11 A4 99 AE  : 1F
0A60  6D 1A 9C D1 77 9A 47 49  : 95
0A68  B7 69 8F 56 97 97 96 FE  : C7
0A70  87 C6 7E D7 3F B7 3F 3F  : 16
0A78  B8 1C FF A5 7F 11 AE 7F  : 35
------------------------------------------
SUM:  5E 3D 5E 3F 0D 30 2A AD   4FB5

0A80  DF 1A 3B 7D B9 F9 FE 3C  : 9D
0A88  BF 4E 5E 7B 79 27 09 9F  : 2E
0A90  6C 99 F3 B0 0C 3B 1D 7B  : 87
0A98  FE C7 95 FF 3E C5 92 EF  : DD
0AA0  3E DF FF 39 FA A6 95 37  : C1
0AA8  F0 38 34 D4 24 BB 33 FB  : 3D
0AB0  9D 27 37 DD D6 A5 A6 FF  : F8
0AB8  EA 18 66 A6 3E A0 E8 F7  : CB
0AC0  F4 EC 2D A3 7C E9 D9 BE  : AC
0AC8  A8 FE C4 AB D1 74 ED E7  : 2E
0AD0  6E C9 28 33 01 32 2D 77  : 69
0AD8  7F 30 6C D3 99 06 98 05  : 2A
0AE0  A4 A9 2D E3 A4 E6 7B BA  : 1C
0AE8  DC 6E 51 9E 25 2B 8F 74  : 8C
0AF0  2D A5 DB A1 E7 C6 69 81  : E5
0AF8  FE 2C AC A1 D4 1B 49 9F  : 4E
------------------------------------------
SUM:  F1 E9 7B 4E 19 4D 53 DC   8D65

0B00  0C 0F E0 B0 17 32 95 5F  : E8
0B08  B1 79 75 9B 33 80 E0 AF  : 7C
0B10  94 67 31 FB 65 0C DF 82  : F9
0B18  D2 8C 79 FE 54 C6 2C 66  : 81
0B20  A8 A6 D6 3F 71 30 6F DC  : 4F
0B28  5B 19 BD D3 5B 27 9F 25  : 4A
0B30  A6 3A 5A 0E C1 C7 75 36  : 7B
0B38  79 9B DC FD 17 E7 DA F7  : BC
0B40  E0 DB 5D B8 1A A8 8B DE  : FB
0B48  91 BF D6 C6 7A 0F 05 CA  : 44
0B50  A6 08 B0 7A 00 F0 C5 BD  : 4A
0B58  20 F5 6E 0E A7 F4 E2 4F  : 5D
0B60  67 56 0F 2F D9 B5 23 85  : 31
0B68  C5 2D 4A 48 F9 58 CD B2  : 54
0B70  E1 0C F7 43 CE CC 5D FA  : 18
0B78  36 AE 87 8E 5D 67 C2 CF  : 4E
------------------------------------------
SUM:  BF E3 F0 AF DF 64 23 D8   692B

0B80  C6 1E 3A 05 A7 87 8E C6  : A5
0B88  0B BB 23 73 A8 47 8E 93  : 6C
0B90  1B 5A CF 30 FD 5B 47 90  : A3
0B98  C3 C7 AD 2F 58 34 EC F1  : CF
0BA0  0F DB 79 5C 82 35 FB     : 67
0BA8  88 6B 2A D1 24 A5 57 FF  : 0D
0BB0  C2 45 A4 B5 67 99 42 21  : C3
0BB8  84 85 26 F3 06 87 E5 1F  : B3
0BC0  46 F2 8F A6 E9 03 38 7C  : 0D
0BC8  AC 0D 26 AD 11 C7 34 AA  : 42
0BD0  47 48 5B 9D 3E 33 78 6D  : DD
0BD8  DC 7F 31 9E 51 5A C6 45  : E0
0BE0  5E 21 56 17 3B 4C AD 0C  : 2C
0BE8  78 B8 D8 B3 01 74 BD 30  : 1D
0BF0  5A 93 C7 E9 3A 48 F9 4E  : 66
0BF8  CB D4 2B D4 A9 D9 E5 52  : 57
------------------------------------------
SUM:  9C 10 A7 C1 5F 50 F4 C8   B8CB

0C00  43 52 D7 FB B9 C6 49 C5  : F4
0C08  36 0A D2 8D 3B C9 3E A3  : 84
0C10  E4 80 8F 25 A0 8D 88 77  : 44
0C18  47 BE 77 A4 6D E9 AF DF  : 04
0C20  39 EC 84 18 5A CA 71 B5  : 0B
0C28  79 27 CB E7 4F 97 AC 55  : 39
0C30  E9 73 DF CB FC 54 90 C0  : A6
0C38  ED 00 DA 29 26 EC 83 BF  : 44
0C40  1C 67 AB 56 B4 47 31 EB  : 9B
0C48  5A 23 BC 5E AC 34 CF DC  : 22
0C50  29 90 4D CF 4C B4 71 7A  : C0
0C58  B7 C6 81 57 D9 C3 76 E4  : 4B
0C60  7D 70 12 7B C6 4E 3C 0A  : D4
0C68  B4 F3 8E D0 8B A3 6C 7E  : 1D
0C70  51 EA DF F4 89 B4 D7 BC  : DE
0C78  A0 94 FF 91 9D D9 25 3F  : 9E
------------------------------------------
SUM:  A4 E1 6A EE C8 16 79 EF   FA15
```

```
0C80  B9 C5 DB FD 70 D2 DC E4  : 58
0C88  43 C1 A7 1C 3F D7 2A 94  : 9B
0C90  6A 43 36 56 80 E6 D7 4D  : C3
0C98  1E C2 12 27 79 6E 90 44  : D4
0CA0  3F 3F 33 AA AA 39 22 C5  : 25
0CA8  F4 37 EC 26 B3 A3 2E DD  : 9E
0CB0  58 08 8C E8 B5 84 78 8D  : 12
0CB8  FD 8A CE 94 78 7E 9F A7  : 25
0CC0  52 9F B2 69 AF 5A 75 20  : AA
0CC8  35 0B CC 2C 6D 86 A0 28  : F3
0CD0  CD 80 16 D5 6E 71 F3 60  : 6A
0CD8  7F BA 82 D2 79 12 C5 D9  : B6
0CE0  D4 25 A0 EF 7B EB 4F AB  : E8
0CE8  8B B3 49 3A DA FA AA C4  : 03
0CF0  8B 5C 25 6F A0 B2 CF E8  : 84
0CF8  1D 1B B9 FA EC 7A DB 0F  : 3B
------------------------------------
SUM:  E6 C6 20 B0 16 4F 44 C6   E2BD

0D00  C3 A8 FB 61 CB 5F 73 5D  : C1
0D08  FC E7 62 CE 6B 69 E2 33  : FC
0D10  16 F3 55 13 57 BC 63 D8  : BF
0D18  D7 5B 07 33 3D 7E 82 F2  : 9B
0D20  C0 70 E0 4B 4B 53 C0 ED  : A6
0D28  67 9C 4A 3B C8 DA 99 D4  : 97
0D30  27 6D 30 97 DF DC F1 57  : 5E
0D38  7E 18 4B B1 CB 54 E1 8B  : 1D
0D40  9A EB 17 8F 1C BD EC 32  : 22
0D48  D1 FB 7D 03 AF 66 E9 03  : 4D
0D50  24 9C 24 6E EE 06 EF 1F  : 54
0D58  55 42 44 70 F9 C9 A7 2B  : DF
0D60  4B 3F 55 33 3E 30 FE 71  : EF
0D68  61 CD 60 66 D3 60 97 AB  : 69
0D70  8E 4B F6 84 49 94 6A 33  : CD
0D78  66 42 0E 16 A0 B8 75 78  : 11
------------------------------------
SUM:  FC CB 13 E6 33 2D 44 43   70E7

0D80  38 51 C1 AF 3A 30 87 66  : 50
0D88  F0 6C ED D8 E0 D7 DD 1D  : D2
0D90  26 4C 6F F7 51 77 36 16  : EC
0D98  A6 6F A9 2B 71 13 A6 18  : 2B
0DA0  8B A7 DC A7 82 4E 76 13  : 0E
0DA8  99 8C 83 FA 7C 55 F1 A3  : 07
0DB0  0E CF 3B B2 DE 94 17 F0  : 43
0DB8  5F BC E8 C8 84 23 D5 BF  : 06
0DC0  41 49 F2 F5 45 1F 44 CB  : E4
0DC8  3D 3B 2A 33 C7 31 1E 73  : 5E
0DD0  58 35 78 91 9C 2C F4 17  : 69
0DD8  CF 3A EA 52 CE C3 EF 47  : 0C
0DE0  77 46 83 76 17 26 7C A8  : 17
0DE8  01 98 65 7B 92 67 15 AC  : 33
0DF0  BD 51 FB F5 47 8D 5F 8B  : BC
0DF8  BF AE B4 F3 CB B0 6F 73  : 71
------------------------------------
SUM:  1E 06 5D A8 6D F4 37 04   FBFE

0E00  42 3C 64 F0 D3 3C 44 4D  : 72
0E08  DB E9 15 07 84 22 A5 49  : 74
0E10  60 49 D8 FC 08 A0 2E 79  : CC
0E18  A2 08 2C 35 90 0E 21 DA  : A4
0E20  85 FC D6 D4 13 B5 B8 59  : 04
0E28  A5 86 4C 55 B3 91 00 18  : 28
0E30  15 19 56 AD 67 72 16 15  : 35
0E38  E3 0F 01 D6 08 CD B7 41  : 96
0E40  05 AA 99 B5 5A 6E 0B 5D  : 2D
0E48  C7 5F 1B 0C 0C 67 46 A5  : AB
0E50  E4 48 3F 9C C1 6F 39 A3  : 13
0E58  9B A3 11 2E E6 F0 3F 51  : E3
0E60  06 37 FC 0E 57 A4 83 15  : DA
0E68  A8 1A E7 94 AD EE 4D 61  : 86
0E70  16 15 87 9D F7 D6 EE E3  : ED
0E78  4B 01 9D 6A 99 8E 98 55  : 67
------------------------------------
SUM:  9B 7B 01 08 C5 BB DC 54   C415

0E80  FF F8 85 7E 60 5B 80 0B  : 40
0E88  A7 3C 2D FA 3F 3A F0 7A  : ED
0E90  CA 55 22 EC DA 02 47 ED  : 3D
0E98  2F F9 6A 13 32 FE 6F 9A  : DE
0EA0  A6 BE 7B D3 D2 BF F4 DD  : 14
0EA8  3B D5 0D CF E0 90 F4 81  : D1
0EB0  A2 0C FB 18 D4 C5 CD 21  : 48
0EB8  9F 99 B2 C6 E7 EE 53 9D  : 75
0EC0  17 FB 88 F9 5C 0D 86 45  : C7
0EC8  0F 6C 6E 7B 17 E5 B7 4D  : 64
0ED0  FA 0A A4 A2 96 DE F8 7F  : 35
0ED8  49 6C F4 71 73 4D 45 BD  : DC
0EE0  83 37 D2 0E 62 72 4C 74  : 2E
0EE8  53 FB A4 8A 78 38 48 01  : 75
0EF0  C8 37 E2 28 19 13 6D BD  : 5F
0EF8  CE DB 52 B7 74 49 C1 8F  : BF
------------------------------------
SUM:  96 DB AB F5 FB BA 6A B7   6614

0F00  5C 1A 7F 86 AC 07 97 A6  : 6B
0F08  87 1E 30 B5 CA 9D A8 5F  : F8
0F10  D0 DF 9B C7 97 6A 51 CC  : 2F
0F18  B9 A1 3D CC FB 51 E0 52  : E1
0F20  10 2F C1 8B BA 79 ED 8E  : 39
0F28  9B E7 28 DD E7 00 80 AA  : 98
0F30  ED 43 E6 64 54 B7 25 C8  : 72
0F38  3F 53 37 63 D0 70 47 63  : 16
0F40  F9 6F E4 FF 35 61 AC 62  : EF
0F48  FE C0 21 05 F2 73 AE FC  : F3
0F50  D2 2C F3 24 AA 14 D3 22  : C8
0F58  97 D3 1C F5 E8 F2 B2 2B  : 32
0F60  01 0E DB 19 2E A7 44 F9  : 15
0F68  A2 4D 38 B5 2E 89 91 33  : 57
0F70  CE 3C 58 31 4A C4 1B 6E  : 2A
0F78  C4 9F 13 B7 73 D0 CB 99  : D4
------------------------------------
SUM:  D8 C8 1F D0 9F 9D E3 64   1D45

0F80  BD A9 66 45 6A 96 D1 56  : 38
0F88  8A 3D 64 FE E4 D0 F0 32  : FF
0F90  E7 3A 5C C8 B5 5C 30 B2  : 38
0F98  99 5B BC 44 ED DC 98 DC  : 31
0FA0  33 B7 76 63 7B 63 B5 71  : C7
0FA8  60 CC 6A 51 7A 76 94 D0  : 3B
0FB0  8E BD 19 B8 32 F8 2B F1  : 62
0FB8  65 93 76 B0 9A 85 67 9B  : 3F
0FC0  67 C9 E9 C3 67 AC 84 52  : C5
0FC8  8C F8 30 CB 59 A9 4A D6  : A1
0FD0  6D FA 11 E5 5F C0 17 38  : CB
0FD8  AE 29 E0 78 DB F5 7B FB  : 75
0FE0  C2 F3 C8 1B 83 13 03 4D  : 7E
0FE8  82 A3 27 0B EA 94 F8 6F  : 3C
0FF0  F6 8E 03 86 75 4A 70 4B  : 87
0FF8  80 63 55 67 90 D1 18 1F  : 37
------------------------------------
SUM:  15 B9 A2 69 1D C0 47 64   1FC0

1000  4E B5 47 7B 8B 83 78 36  : 81
1008  D3 FA 2F 04 7A EC 31 C7  : 5E
1010  41 CC D7 6A 6C 2B F0 FF  : D4
1018  7C 98 5B 8F 59 A7 A9 65  : 0C
1020  79 F9 61 8C 1C C1 04 C0  : 00
1028  86 A4 32 2E F4 7B E5 66  : 44
1030  7B F6 AA B9 90 5E F9 09  : C4
1038  9E F6 C4 C0 29 68 D8 A3  : 24
1040  DD A3 25 7C B8 1C C0 06  : BB
1048  56 66 C8 CC C2 F5 EA CF  : C0
1050  9F 96 7A FC AC EA CE 68  : 77
1058  86 7C DC D7 DE 10 A7 97  : E1
1060  32 FF 3A 10 07 F8 C7 23  : 64
1068  DB 92 1E 4D C0 4C 37 FA  : 15
1070  0C ED D6 5E 67 41 DD 17  : C9
1078  38 92 C9 D6 B5 AB FA 39  : FC
------------------------------------
SUM:  9F C7 E3 57 7A 7E F0 74   D6B5

1080  45 E1 88 0E FE 2C 7C B4  : 16
1088  1C B0 07 D2 1B 1A 00 E4  : BE
1090  EA 4F 23 F4 F0 CC E2 9C  : 8A
1098  0B FB 58 36 C4 59 39 2B  : 15
10A0  57 6F 8C 4F 1F B5 B2 66  : 8D
10A8  69 3E F9 54 A6 2E 06 6B  : 39
10B0  D0 68 FA 23 49 AA DD 37  : 5C
10B8  03 17 84 2E B7 69 E1 18  : E5
10C0  26 69 CE 91 5B 66 AB 69  : C3
10C8  97 D1 C0 B4 C7 94 AA 22  : 03
10D0  CE 90 4E 7F 5F D1 28 F5  : 78
10D8  4D CF D1 A7 8C 11 F2 DC  : FF
10E0  96 C5 4B 97 62 52 11 44  : 46
10E8  4A 78 79 AC 5D 99 8A 8A  : F1
10F0  61 59 7D 1F 26 AD 4C 47  : BC
10F8  7B 43 1B DC 33 7A FC 47  : A5
------------------------------------
SUM:  7D 79 16 A7 B7 4F 5F 37   004E

1100  83 A3 ED D3 A7 67 FA 93  : 81
1108  9C C8 88 CE B5 0E 0F 1E  : AA
1110  46 9D 59 E8 F5 8F F5 49  : E6
1118  76 E8 EE 4D 48 0A BD F0  : 98
1120  43 DD E9 D4 B3 7A 44 D7  : 25
1128  18 D9 ED 01 58 D0 BE F5  : BA
1130  AA 4D E0 B5 6A 7D BB 42  : 70
1138  7E 03 9F 8B 6A D3 EA A8  : 7A
1140  65 72 24 97 D7 31 E7 02  : 83
1148  3F 6F 4B 22 38 36 9D 6E  : 94
1150  59 78 CC 03 D3 17 9D 13  : 3A
1158  EF C2 00 EA 3B 99 81 EC  : DC
1160  26 2B F4 F9 11 B7 F9 13  : 12
1168  2B 42 96 E4 97 9A DB 56  : 49
1170  5E E4 5C 38 D8 80 0F 37  : 74
1178  96 68 0E 3B 7B D1 1D F5  : A5
------------------------------------
SUM:  8F CA 40 E1 90 61 04 A4   09E9

1180  A1 80 36 7F AA C6 BE 60  : 64
1188  5E 99 57 AD 05 F0 30 75  : 95
1190  03 4F B4 92 7B 73 2C E7  : 99
1198  0F BD 26 BF 51 72 44 32  : EA
11A0  4C A8 86 C2 57 D7 DD 83  : CA
11A8  75 76 FD 0D 06 17 8B FB  : 98
11B0  ED 2D DF B2 A3 23 79 A4  : 8E
11B8  8B ED 91 A7 B1 E9 03 81  : CE
11C0  C4 B3 6C E7 23 71 89 65  : 4C
11C8  47 7E 90 7B 17 86 6C F3  : CC
11D0  8C 8D C7 0C C5 1A 0F FC  : D6
11D8  00 FB F3 81 7C E9 80 58  : AC
11E0  47 02 F9 D3 00 B0 80 16  : 5B
11E8  0A 10 49 74 39 21 1D 7D  : CB
11F0  B1 21 D6 16 96 BB 67 ED  : 63
11F8  13 0F E8 FA 77 02 41 2C  : EA
------------------------------------
SUM:  F6 58 10 EB ED 1D 0B E9   70F3

1200  FC 93 42 9D 82 47 2F 89  : EF
1208  2C 0C CB B7 71 EB 1E 1C  : 50
1210  6E C1 1C 2C AF 44 B1 34  : 4F
1218  CE E9 CB 83 4A 0E B8 33  : 48
1220  FA 51 97 70 59 7E 7F F3  : 9B
1228  1F 2E 76 37 7D AB B5 1A  : F1
1230  BD A7 91 38 79 19 7C C6  : 01
1238  A2 8A DB 9B B2 BD EB FA  : F6
1240  A2 70 B4 88 BD FE A9 FD  : AF
1248  3B 37 B2 25 52 CA E8 A8  : F5
1250  36 5D C4 BD 54 BE EC 4F  : 61
1258  7A D9 4B 5A 5C 71 0B 47  : 17
1260  65 2F 37 D0 67 F0 D3 64  : 29
1268  FC BB F3 0E 98 94 3F 44  : 67
1270  AB 98 13 92 09 5B 37 B2  : 35
1278  3C C3 20 90 29 D2 EA 93  : 27
------------------------------------
SUM:  B1 1B 3F 41 DD 2B 0C 01   F5A7

1280  20 49 6D 9C E2 13 C9 F6  : 26
1288  E6 12 6D C8 AB D0 79 86  : A7
1290  23 1D 04 76 63 70 51 6D  : 4B
1298  76 08 E6 36 21 59 FC 3D  : 4D
12A0  D9 6E FE 81 53 EC 03 3E  : 46
12A8  FB 49 18 85 5C B5 72 EF  : 53
12B0  10 13 35 D0 2C 72 04 A3  : 6D
12B8  B0 A0 DC 93 99 F3 92 5A  : 37
12C0  2F 50 2B 8F F3 F5 9C 04  : C1
12C8  34 78 5F 83 C8 6A 48 6B  : 73
12D0  5D 67 47 17 82 7E B5 F4  : CB
12D8  26 75 AE 40 FD 6B E8 4C  : 25
12E0  EB 5C 80 EB 53 CB 61 5C  : 8D
12E8  42 EF A2 83 84 0B 6B D1  : 21
12F0  40 7B 1D 89 92 53 F4 BE  : F8
12F8  C6 CF B2 FF 71 5E 9F A9  : 5D
------------------------------------
SUM:  4C 23 5B D8 99 81 7A 93   B821

1300  F6 69 65 DE 9C 5B A5 1D  : 5B
1308  52 8B 00 DA FD F2 DF A0  : 25
1310  AB 68 D5 A0 1A 75 B9 6B  : 3B
1318  19 8B F5 BD C3 5C 1A 8C  : 1B
1320  B5 7B 7B B9 98 38 61 95  : 2A
1328  D5 42 29 73 47 47 6F 05  : B5
1330  24 32 E1 95 73 21 47 0E  : B5
1338  45 64 51 81 68 B8 C9 CE  : 32
1340  7B 76 3A 80 78 66 6D D4  : CA
1348  B1 00 30 E4 9D 67 A1 CA  : 34
1350  32 0B 4F F4 AE FB 93 36  : F2
1358  2B 55 79 3D 91 F7 A9 6B  : D2
1360  B9 83 FA 65 9D 48 3F 53  : 12
1368  8D 27 4D 46 5D 6A 3A F7  : 3F
1370  FB C7 34 DF E0 27 22 03  : 01
1378  1B EE 70 1F 5F 04 E2 9A  : 77
------------------------------------
SUM:  E4 6F 22 95 BD 12 FE 50   66DA

1380  43 67 4E D2 4E DF E9 DC  : BC
1388  31 74 73 86 06 AF E3 B4  : EA
1390  92 F2 EA 3F F2 DE 79 DE  : D4
1398  86 8B 0A 37 FA 74 68 52  : 7A
13A0  EE EE 66 EF A4 54 49 33  : A5
13A8  21 B8 7E 74 A6 0E 86 BF  : C4
13B0  5B 63 C1 A1 B8 C7 B4 B0  : 03
13B8  57 58 0D 80 B6 A1 99 93  : BF
13C0  B8 26 E5 5A 0A 9F 64 D4  : FE
13C8  0D 93 49 D4 D0 F7 1F 36  : D9
13D0  99 A8 3B EC FE DC 6C 42  : F0
13D8  D8 E2 90 27 6E C5 D1 C9  : 3E
13E0  45 68 07 DB 27 50 DC 5D  : 3F
13E8  27 3A 8A 00 38 9E 81 2C  : 6E
13F0  84 B2 91 2A FC E1 D2 E5  : 85
13F8  10 0F B9 99 A4 EA 15 CD  : E1
------------------------------------
SUM:  83 5F 3B 31 3D 9A CD 45   EC9B

1400  8D 12 4E 90 A9 FF 11 0F  : 45
1408  CB B9 FF 90 86 A3 E6 D7  : F9
1410  E0 36 44 16 A3 DE 8D B9  : 37
1418  1B 9E 1B B7 2C 5E 01 EF  : 05
1420  AA 6B DA 6E 8E 0D 30 B2  : DA
1428  96 9A 8B C5 D4 E0 4C E7  : 67
1430  6A 78 41 A6 4F 07 08 47  : 6E
1438  A2 11 BF E8 53 9C 11 A0  : FA
1440  CC DA 78 8E 1C A7 85 9D  : 91
1448  99 50 85 1C B1 DC FD 91  : A5
1450  59 50 D4 ED B2 77 1E ED  : 9E
1458  A8 47 30 B4 B5 5F 1A 0C  : 0D
1460  B4 92 12 CD 9B C6 27 91  : 3E
1468  6D DB 94 C2 1A CE D2 45  : 9D
1470  73 CC 3C C4 2D F7 5B B0  : 6E
1478  85 74 5E 12 03 4B 4C A4  : A7
------------------------------------
SUM:  1E 9B 52 5E 1B 9D 74 5F   6436

1480  DC 32 96 C7 15 7D 88 5C  : E1
1488  C8 53 3E 12 D9 20 D6 2F  : 69
1490  C7 1F 2C F3 72 6A 98 E4  : 5D
1498  BF 9C 25 AF 29 47 62 CD  : CE
14A0  D9 EC FB 59 0F 23 1B BE  : 24
14A8  E5 B4 D6 7C 4E 6A 0D 82  : 32
14B0  31 E9 05 CE D0 6E FF 91  : BB
14B8  4A 21 1C 87 AF 0A 8B 2C  : 7E
14C0  4F 5E EA 70 37 8E E1 F3  : A0
14C8  C5 10 7B A5 51 C0 B3 D9  : 92
14D0  1E 3B B8 91 33 7E DD FD  : 2D
14D8  16 04 FA F0 9E 7E 7C 5E  : FA
14E0  F6 6A 6B CE C0 9B 84 FB  : 73
14E8  01 C6 2E E6 C2 C3 13 28  : 9B
14F0  DF 57 68 6A 92 CF 77 B9  : 99
14F8  AC FD 96 7F C0 49 5B 6C  : 8E
------------------------------------
SUM:  2D 1B C5 D8 92 13 60 A8   8CD9

1500  FA 56 39 0C 8C 28 50 92  : 2B
1508  B1 AA F4 FF 27 AB EF 6D  : 7C
1510  BD A4 B4 58 EA 78 50 8D  : AC
1518  55 6D 27 C0 CF F7 AB 66  : 80
1520  F4 09 76 39 2C 8F 69 4D  : 1D
1528  91 C8 D3 64 72 54 D9 28  : 57
1530  49 48 8B 12 3D 84 18 BA  : C1
1538  1B 3B 71 17 50 D5 E3 86  : 6C
1540  76 1C 2C 57 AA 63 6A B9  : 45
1548  B8 A0 8B 7D 57 26 C8 CD  : 72
1550  FC AA CE 43 63 62 BD 71  : AA
1558  EB E9 A5 48 6C 7E 28 14  : E7
```

▶ 先日，車を運転していたら後ろから追突され，首がむちうち状態になってしまいました。おかげで友人からは「首はむちうち，腹はむちむち」といわれています。

日高　のり子(21)奈良県

```
1560  9E AB 74 7A 77 63 DE 11  : 00
1568  FC 77 A7 A7 D5 C5 7E 01  : DA
1570  4F E1 AC E4 19 17 F8 38  : 20
1578  F2 B7 7B 93 32 CD F1 4B  : F2
-----------------------------------
SUM:  96 6E B9 E0 FE F3 D3 47  8CFA

1580  73 70 6C 55 E0 B9 E0 24  : 41
1588  0B 76 97 85 74 2C 51 35  : C3
1590  9B 92 85 92 89 A3 AF 76  : 95
1598  D6 1D 73 E0 F8 27 A7 98  : A4
15A0  97 80 43 D4 91 CC F4 EF  : 6E
15A8  79 4D 5B D8 DC A6 BC 75  : AC
15B0  E3 3E 59 02 A4 38 6F CC  : 93
15B8  EA 01 7D D3 8A 78 84 29  : EA
15C0  4A 60 A5 23 6F C7 A9 B0  : 01
15C8  BD 2C 3F 08 F4 F1 0F B7  : DB
15D0  77 1B 1D 64 16 D0 70 AC  : 15
15D8  C6 B0 D7 46 5C 20 B3 C5  : 87
15E0  A7 21 5F 04 E3 1F 77 15  : B9
15E8  BE 14 F1 AA 86 D9 4F 68  : 83
15F0  97 E0 1F 77 31 72 91 2C  : 6D
15F8  1A D1 AF CA 9E 55 8D CB  : AF
-----------------------------------
SUM:  26 DE 65 91 7D 38 E9 0C  646A

1600  CF 6D 67 2F 3D B5 6F 54  : 87
1608  E6 9E DF 94 7B 7E 60 20  : 70
1610  D4 1F EF 86 A5 BD 53 E3  : 00
1618  4F 60 50 62 F5 5B 74 C7  : EC
1620  68 EB 18 26 E8 F5 6D C9  : A4
1628  D4 FA 5D CA 72 A0 B6 66  : 23
1630  AE C7 5C 8D E2 37 70 BD  : A4
1638  C8 F1 36 D7 06 9D 57 E1  : A1
1640  2D 6E E3 AA 4A E0 C6 F0  : 08
1648  BC 94 DF 08 7C F7 A5 93  : E2
1650  F0 BC 9E 9E 4D 03 1B 04  : 57
1658  B2 03 16 D8 1C 1E 40 56  : 73
1660  DC 8C D6 EE 18 61 34 B3  : 8C
1668  E7 49 3F 67 7D 63 AE 28  : 8C
1670  A1 E6 DE 25 DE 75 D0 CC  : 79
1678  61 F5 0E BA E0 AF 4C 8D  : 86
-----------------------------------
SUM:  DA 98 03 5B 16 94 44 FC  6314

1680  6F BC DE 71 7F D9 4F CA  : EB
1688  33 81 6A EF 0D 10 CF 62  : 5B
1690  72 0D 41 33 41 F8 C7 CE  : C1
1698  D0 1E 9D F1 77 2F 85 E3  : 8A
16A0  90 55 A2 95 EC F9 56 3A  : 91
16A8  E2 76 9F B0 81 93 A9 67  : CB
16B0  F3 B3 8B EE 9A F5 A1 1E  : 6D
16B8  BC C2 F5 83 92 2E 01 3D  : F4
16C0  37 C7 A4 EF 88 B6 F1 88  : 48
16C8  B2 DA 4D FE 33 62 51 F7  : B4
16D0  E7 2F 54 72 55 B7 33 3D  : 58
16D8  BD D7 A4 5D 16 B3 8C 9B  : 85
16E0  66 6B 5A 64 EF B4 38 F7  : 61
16E8  B4 9F 20 DE 32 75 47 25  : 64
16F0  5D 2F 19 9C 1D 8E B8 FA  : 9E
16F8  27 C1 99 7B C7 37 0C 8F  : 95
-----------------------------------
SUM:  30 49 FC 4F 08 2F 4F D5  396B

1700  7F 59 A7 2E 1D D7 69 DF  : E9
1708  FA A2 C5 71 B2 57 7A 3A  : 8F
1710  4D 2E F2 33 F1 E7 BC 2F  : 63
1718  34 D7 3F 4F F3 5E 3E 1C  : 44
1720  1F 74 29 B1 76 D8 3B 9B  : 91
1728  CD 14 A4 B7 2E 3A EF FD  : 90
1730  50 ED 1F 5E B3 D1 4B F5  : 7E
1738  68 45 FE 44 BC 56 C7 3D  : 05
1740  BC 53 98 3D 69 4F 7B 00  : 17
1748  BD 5E 87 57 D6 A0 3D EB  : 97
1750  D3 DA CF 47 B9 38 2D C9  : AA
1758  02 D5 E1 A7 41 17 8F 0F  : 55
1760  C7 81 DE B3 7A F0 B0 B8  : AB
1768  D5 8A 5D 3C 16 A1 D7 E3  : 69
1770  28 EC AB 96 A1 3F 8E 36  : F9
1778  CA AD 5A 1B 25 08 A2 E9  : A4
-----------------------------------
SUM:  7A BE 96 4D 55 C2 44 AB  1149

1780  78 95 BE F4 EB 78 16 52  : 8A
1788  17 52 7A CA B8 F6 3C 25  : BC
1790  3E 12 B3 8C 16 26 CE 5D  : F6
1798  E9 E7 B4 EB B4 F1 D7 39  : 24
17A0  B8 F3 8E 52 F7 D5 BE F6  : 0B
17A8  08 D5 DA 11 BB D2 1A 7D  : EC
17B0  EB DA 05 90 D2 94 A2 B6  : 18
17B8  EB 72 E3 36 AD F7 81 00  : 9B
17C0  E8 6C 75 D5 DA 14 E5 26  : 97
17C8  C3 DA 0D 98 88 36 24 9A  : BE
17D0  BA B3 9E E0 4A A3 6C 23  : 67
17D8  C2 CA E0 4B E0 30 A7 1B  : 89
17E0  D9 16 E5 95 17 79 D5 5C  : 2A
17E8  7E 59 5C 59 48 4A 0C E4  : 0E
17F0  E8 50 6C 83 7C 19 B0 66  : D2
17F8  40 12 AD 2F E1 1F 78 11  : B7
-----------------------------------
SUM:  F2 88 49 96 E6 CF 17 EB  2A0C

1800  A4 65 CD 37 F3 2C 86 86  : 38
1808  1B 11 07 88 31 DC 5D 55  : 7A
1810  6E A3 CD 63 6B E1 DB EC  : 54
1818  3C B1 0F AF 78 C7 90 4F  : C9
1820  CA 4A CB 17 FC 2B A3 4B  : 0B
1828  B1 0D 19 14 41 2F ED C0  : 08
1830  08 89 BF 63 3B 50 6C DD  : 87
1838  A5 94 9A 81 14 4E 08 2C  : EA
1840  D7 01 E1 F4 0B C4 BB CF  : 06
1848  C2 DD 15 3D A6 60 FD DF  : D3
1850  B9 26 C5 4F BC DC 52 8C  : 69
1858  1A 6B A7 F7 31 3F 5C D3  : C2
1860  8B 1D 23 79 37 51 97 CE  : 31
1868  72 D4 79 89 B8 33 A3 4E  : 24
1870  4D B1 CE B7 4D 7B 1D 75  : DD
1878  6F BC 90 DA A7 27 BD AE  : CE
-----------------------------------
SUM:  B6 0B 49 EA 14 0D CC 76  1747

1880  01 2E C1 E2 57 7C A4 48  : 91
1888  A9 AE 78 69 B9 57 FD 72  : B7
1890  D3 C9 44 4F 61 17 8E BC  : F1
1898  3A ED 78 32 16 D1 D7 24  : B3
18A0  26 99 2B 4F 7C 4E 2F 05  : 37
18A8  97 5D BB 6B DE CE 80 41  : 87
18B0  15 70 D4 FC 3A ED EF 5D  : C8
18B8  29 03 47 F4 8A F5 35 EB  : 06
18C0  B4 32 59 AE D0 D7 FC 4C  : DC
18C8  AF 07 AA 1B 0F 56 CF 69  : 18
18D0  61 AE 0E E5 E9 46 89 79  : 33
18D8  55 F1 1B 31 1B AC 59 D4  : 86
18E0  96 63 AE 8F B8 11 82 E1  : 62
18E8  34 53 26 89 46 56 B7 E2  : 6B
18F0  C1 03 98 47 DC 58 98 41  : B0
18F8  A7 0D CC FC 61 6C 62 B4  : 5F
-----------------------------------
SUM:  FD 99 5A B0 C3 03 B9 E2  22E0

1900  7D D5 6F 4A F6 70 5E A4  : 73
1908  80 CA 9F D5 C8 F5 B2 33  : 60
1910  67 81 16 2E 92 3B F3 30  : 1C
1918  FD 61 0F 0C C6 BD F3 C9  : B8
1920  22 66 6D 38 A9 3E 63 83  : FA
1928  4E 52 94 D3 71 A6 F7 F3  : 08
1930  A1 90 16 8F 1E 0B 76 9F  : 14
1938  CF C9 D1 35 24 CC 5C EE  : D8
1940  CB 68 84 20 9C E8 72 1C  : E9
1948  21 63 E5 40 8A 90 B3 59  : CF
1950  D0 14 E4 4E 84 7F 13 17  : 43
1958  0C D3 E9 75 78 88 66 7A  : 1D
1960  D0 CD 39 42 38 AC 33 A2  : D1
1968  2E 8D 8D C9 22 2D 06 C6  : 2C
1970  C2 2B 5C 97 56 9A D0 4C  : EC
1978  CD 5A F7 06 4B 27 0C CE  : 70
-----------------------------------
SUM:  96 23 6A F3 8F 31 D5 5B  324D

1980  3B 51 76 CD 5F 75 AE 53  : A4
1988  1E 30 A7 45 4C 34 BA 68  : DC
1990  A7 05 62 DC 08 DF 81 09  : 5B
1998  93 DA 99 C0 AD 0C 5C C4  : 9F
19A0  55 E2 E2 19 DA 74 A6 C4  : EA
19A8  0E 49 74 3A 1E 94 D3 EB  : 75
19B0  3A 69 09 30 FD D0 5C 62  : 67
19B8  B8 88 D8 6B 8B 9E D3 62  : E1
19C0  CB BB 0B AB CF AC FC 53  : 06
19C8  DF AD 8E 14 D9 2B 57 65  : EE
19D0  18 1C 0E 31 54 2C 93 D2  : 58
19D8  DC 44 70 A4 2D 2D C4 37  : 89
19E0  2A 52 D9 63 E6 5A D3 69  : 34
19E8  6E 1B 19 4A 0D 28 63 45  : C9
19F0  47 B6 54 53 2D 67 F4 B7  : E3
19F8  1D B9 94 71 A5 B8 B0 07  : EF
-----------------------------------
SUM:  82 20 40 A1 CE DB 71 28  DE3B

1A00  0A B3 82 ED 52 10 BC A0  : EA
1A08  45 B2 E0 CC 40 67 34 B1  : 2F
1A10  30 2F A0 55 F0 32 57 4A  : 17
1A18  BC D2 DC 20 65 38 63 CF  : 59
1A20  08 AB 9D 2D C3 83 29 EC  : D8
1A28  9C AA 78 61 D1 7F 24 3D  : D0
1A30  1F E4 8E 0F 04 16 26 92  : 72
1A38  F9 24 A1 79 17 35 5C A6  : 85
1A40  78 E5 D9 E7 F1 E7 17 8A  : 96
1A48  6B 95 EB 4A DD 83 D1 4E  : B4
1A50  0B A4 A1 8C 1F 92 7E FD  : 08
1A58  EE DA 3F 78 00 A3 7C 2D  : CB
1A60  A5 B6 63 6E 37 ED 63 16  : C9
1A68  B5 C7 EA 99 F4 A1 85 3C  : 55
1A70  A3 65 40 4A 3D A7 DD 59  : AC
1A78  3C 6B 0A 5E B5 9A 7E 2B  : 07
-----------------------------------
SUM:  0C 08 5D 28 A0 9C 9E A3  A275

1A80  09 49 A4 34 6F B8 D2 88  : AB
1A88  43 8E E8 16 87 BC B9 D7  : A2
1A90  DC C8 5E D9 D4 18 6D 36  : 6A
1A98  D7 12 CC 88 46 BB 43 50  : D1
1AA0  91 8C D1 1C A5 83 28 80  : DA
1AA8  2A 39 79 C0 59 BC 7D 4C  : 7A
1AB0  89 56 7C DA ED 09 D5 CB  : CB
1AB8  B8 22 31 5A 8B 6C 09 BE  : 23
1AC0  19 AC E2 C9 75 F9 7F 7F  : DC
1AC8  80 49 C7 E2 24 DB 5F A8  : 78
1AD0  51 FB 58 30 2B 65 96 60  : 5A
1AD8  E0 C9 6C 3F 97 68 38 BE  : 49
1AE0  50 94 76 6E 1E D1 FE 0F  : C4
1AE8  80 B8 8D CF 83 98 B7 97  : FD
1AF0  4D 2E 4F A0 67 FF 1D 3F  : 2C
1AF8  9A 0F AC 03 D3 93 97 5D  : B2
-----------------------------------
SUM:  7C 30 18 B5 BC 97 D3 C1  65C8

1B00  BA E4 BB FA 75 DA E1 32  : B5
1B08  D5 71 97 B6 9E A7 5E 0D  : 43
1B10  B0 61 31 C1 5A D3 56 8D  : 13
1B18  6E 82 58 9C 03 2B 81 96  : 29
1B20  09 17 45 FC 67 A0 DC 42  : 86
1B28  45 5C BD EB 03 33 3F 0C  : CA
1B30  8B 92 DE 25 D3 2D DB D6  : D1
1B38  FC 46 C1 19 DD 34 3A 60  : C7
1B40  61 F2 34 EB 3C 38 B5 82  : 1D
1B48  3A 9A 1B B7 66 1D A7 C9  : 99
1B50  4E 7E 37 EF 24 5B F6 7A  : E1
1B58  6F 8D D6 B4 41 D5 5E B7  : B1
1B60  DE 87 0C 80 78 26 FD A8  : 34
1B68  50 3D 30 E5 FB 9A 34 F4  : 5F
1B70  C9 E9 FF 67 5C 95 39 9D  : DF
1B78  AC EE 9F 5A 70 5F 3A 3E  : DA
-----------------------------------
SUM:  7D B5 B2 9D D0 EC 9A D9  C105

1B80  9F 52 BF E4 0D 5D 9A 83  : 1B
1B88  B7 FE 46 78 C9 39 5F 4C  : 20
1B90  45 D3 0F 86 11 17 FB 50  : 20
1B98  3B C6 62 16 16 AE 82 BB  : 7A
1BA0  81 35 1A 24 21 82 6E 9A  : 9F
1BA8  35 FB 4C C1 D1 AE 06 B6  : 78
1BB0  13 11 8F 04 B4 89 24 95  : AD
1BB8  D9 78 97 DF 9D 24 7D 3C  : 41
1BC0  19 56 58 38 5E 56 06 0C  : C5
1BC8  22 54 21 FC B8 BD B3 63  : 1E
1BD0  38 7F 60 F2 ED 2D 44 DE  : 45
1BD8  3D 1A 06 52 2F 9E 75 8B  : 7C
1BE0  39 D7 CA 58 77 7E BE 51  : 36
1BE8  5C 83 B9 70 93 68 B1 18  : CC
1BF0  1A A5 57 37 7F 11 81 C6  : 24
1BF8  82 03 41 2A A8 06 BB BC  : 15
-----------------------------------
SUM:  59 E7 FC 61 A3 13 A8 BE  01C1

1C00  A3 6F 16 AB 7B 79 16 1E  : FB
1C08  FE AA 2C 4D DE EF 7F 12  : 7F
1C10  2C 3A 34 01 D6 D4 F8 23  : 60
1C18  BF C7 6A 99 AC 5A DC A2  : 0D
1C20  2A 98 51 3C EB C5 67 A1  : 07
1C28  43 F3 AE D5 42 50 C9 C5  : D9
1C30  BB AE D0 C6 95 9D 08 89  : C2
1C38  CE 58 55 AF 65 5B 56 08  : 48
1C40  1C 74 3B 8A B9 4A 43 AB  : 46
1C48  5E AE 55 95 41 58 C0 34  : 83
1C50  54 F6 F1 2C D0 5E F1 38  : BE
1C58  71 61 97 B9 56 BD E4 26  : 3F
1C60  F3 6C 40 26 21 79 CC 15  : 40
1C68  B0 42 BD B2 EB 31 9A 63  : 7A
1C70  6C 48 9D 35 A6 E7 1E 6D  : 9E
1C78  2C 45 BD B8 9C E3 45 7F  : 29
-----------------------------------
SUM:  FC 5F 73 E1 70 D4 98 8D  8A1A

1C80  0E 04 4D B6 E4 95 8D 67  : 82
1C88  3D 01 B4 54 DE C2 48 DE  : 0C
1C90  FF F0 FB FE 52 88 97 AF  : 08
1C98  0E 31 7B D8 26 56 2B DE  : 17
1CA0  63 CC FD D6 F2 F0 6D 7B  : CC
1CA8  B8 B1 13 89 EC CA 4B 07  : 0D
1CB0  AF 63 38 DF 7F A9 AE D0  : CF
1CB8  99 B3 76 F3 83 F3 89 9F  : 53
1CC0  A7 7A 9F 9C FF 7F EC 17  : DD
1CC8  FE BF 21 FF FF D3 0D B7  : 73
1CD0  07 74 1A F1 08 4F F9 08  : DE
1CD8  F0 52 40 92 56 44 BE FC  : 68
1CE0  11 4C 00 4F F6 D3 F3 09  : 71
1CE8  3D DE BE 52 AE A4 30 AF  : 5C
1CF0  31 35 D4 6E 44 AA 48 43  : 21
1CF8  04 A7 0D 72 25 37 4C 02  : D4
-----------------------------------
SUM:  DA BE EE B0 83 C8 ED 92  30EB

1D00  53 10 AD DB 00 1B 6C E3  : 55
1D08  DA B5 CF D8 B4 39 C2 20  : 05
1D10  7C D8 4E 42 F3 3B 5D D6  : 45
1D18  73 96 08 4F 30 5E E4 4A  : 1C
1D20  F2 46 3F 3E 65 EF 0E 42  : 59
1D28  29 C5 4F 4B 78 92 63 B9  : AE
1D30  EF 21 21 16 1C 26 F3 48  : C4
1D38  54 96 2E 45 13 FA 86 27  : 17
1D40  0D F9 80 8B FC CC D7 95  : 45
1D48  30 8B FC C3 91 7E E8 11  : 82
1D50  7D 6F FD 48 BD A1 5C 0E  : F9
1D58  A1 1A ED F5 1F 61 BC 85  : 5E
1D60  FF 59 19 A0 76 75 FB 3D  : 34
1D68  D1 D1 F7 AF 25 6F AA 77  : FD
1D70  6A FB 3B C4 A1 C0 09 49  : 17
1D78  A5 DE BF CC BC F6 CB FA  : 85
-----------------------------------
SUM:  B4 05 1F 92 44 74 A9 BD  F5AA

1D80  7D 65 95 39 35 E3 4B EE  : 01
1D88  BF 8E 4E 9E D2 F0 42 B7  : F4
1D90  5F 35 77 64 F6 6E 5E 79  : AA
1D98  13 E0 F8 29 E1 89 7F E9  : E6
1DA0  AE F1 F0 6E 1C 48 ED D0  : 1E
1DA8  36 3F 8C 2B 4A 31 4D 45  : 39
1DB0  9A 4D E2 1B AA 85 36 CD  : 16
1DB8  E1 EC DD 95 9D DD E9 9D  : 3F
1DC0  6F E2 46 7D 77 7B 3C 8C  : CE
1DC8  64 D6 6F 56 BC 80 A1 FC  : D8
1DD0  BE 64 C1 AF 2C 6E 48 EC  : 60
1DD8  55 9A 4F 83 5D 5F BB BD  : F5
1DE0  97 C1 FC BF BA 1B 91 77  : F0
1DE8  BF 95 68 86 BF D2 27 45  : 3F
1DF0  6C DB C9 66 B7 DE 30 26  : 61
1DF8  32 43 49 F6 8E F4 93 B6  : 7F
-----------------------------------
SUM:  E7 9B C8 53 05 2C 1E 4F  5425

1E00  13 BA 3D C4 9F 9E 0C 97  : AE
1E08  7E CA E0 8E 49 61 7C 67  : 43
1E10  74 A6 BE 0B 31 A4 7F 4A  : 81
1E18  09 9F 34 16 4A 18 E7 41  : 7C
1E20  70 E4 73 16 79 C8 11 2E  : 5D
1E28  B5 16 9C 29 2C BF 87 8A  : 8C
```

▶夏休みにバイトして「ゼクセクス」の基板を買った。友人にそのことを話したら「あとはゲームを作れば，あんたすごい人になれるよ」といわれた。少し，プログラムの勉強でもしてみますか。
松本　直貴(17)鳥取県

```
1E30  E4 93 6C 7A AE 32 9B F3  : CB
1E38  E1 E9 52 88 1A 11 19 75  : 5D
1E40  5D BF A4 05 63 CB 0C 8E  : 8D
1E48  88 2E B8 B1 05 6F AE 8C  : CD
1E50  1A 8A 43 5F A9 19 F4 E0  : DC
1E58  45 53 75 B0 DF 13 54 7A  : 7D
1E60  AE 21 EC 61 29 63 33 D4  : AF
1E68  09 73 B5 AD 09 ED 2A D3  : D1
1E70  94 19 3F 71 1D 72 F7 4A  : 2D
1E78  1C D0 28 67 A6 20 82 03  : C6
------------------------------------
SUM:  A3 86 F8 5F B5 CD 12 11  9248

1E80  A9 0C BA B8 2B 37 E8 2C  : 9D
1E88  E0 AD FE E8 89 E2 23 2C  : 2D
1E90  B9 3C 01 DE 36 53 92 21  : 10
1E98  0B BA 69 90 78 2D 9A 7A  : 77
1EA0  87 3F D7 C6 F2 82 BC 68  : FB
1EA8  BD 86 95 CB F8 19 E6 62  : FC
1EB0  8B C5 F1 30 EE 5D 67 E1  : 04
1EB8  78 C2 3E 8B 89 3D 7E BE  : 05
1EC0  23 5C 4F DD 6B 99 AD 8B  : E7
1EC8  C7 9C 6E 9E 9F EC F5 F4  : E3
1ED0  C7 00 CC 59 9B A6 96 67  : 2A
1ED8  57 EE C7 D3 DF AC C7 90  : C1
1EE0  93 93 AC BC 42 A9 67 0A  : EA
1EE8  33 ED 8E CA 88 5D BC 30  : 49
1EF0  D3 4C 02 CA D7 06 5B DC  : FF
1EF8  AB 14 52 F1 67 09 F9 86  : F1
------------------------------------
SUM:  E0 C1 9B 42 4F BA 34 6E  6746

1F00  70 21 83 B8 CE D1 88 4F  : 42
1F08  E6 46 EC 42 7F 29 E6 AE  : 96
1F10  95 7E 2F C2 11 C3 0D F3  : D8
1F18  CD 08 EC 75 C9 62 54 77  : 2C
1F20  06 79 A6 59 72 98 E2 5A  : C4
1F28  CC 32 42 4B 2D 56 24 A0  : D2
1F30  86 C6 79 FB 3A 6B 66 FB  : C6
1F38  3A 68 2D F6 94 96 CD FC  : B8
1F40  6A 5B 66 FE 3A 7E 12 7A  : 6D
1F48  54 F4 DD B0 61 A6 C9 E2  : 87
1F50  0E 0D D6 2D CB AF F8 9B  : 2B
1F58  E7 92 C1 08 32 62 4C 7A  : 9C
1F60  34 2A B9 43 78 6E B0 20  : 10
1F68  46 B4 30 7F 8B 55 C5 90  : DE
1F70  73 36 C8 CE 6E 30 2E EB  : F6
1F78  B4 22 E7 1A 7F 3C 4A 9C  : 78
------------------------------------
SUM:  9E EA 8A 53 1C 72 14 00  B522

1F80  91 E7 8C 3F 9D 12 9C 21  : AF
1F88  E6 91 FC E7 D4 E0 0F 3C  : 59
1F90  51 FC FA B5 37 BA 26 47  : 5A
1F98  AC 23 FA D0 29 40 3C E9  : 27
1FA0  55 7A FF AD CE 39 AB 94  : C1
1FA8  40 3C 83 64 68 56 33 97  : EB
1FB0  AA 3E 1B A4 16 B9 49 23  : E2
1FB8  CD 3B 03 37 F9 D3 4E DF  : 3B
1FC0  64 03 73 A3 23 4E AD 58  : F3
1FC8  C5 F5 A5 89 9D E4 9C 02  : 07
1FD0  FB 00 03 16 AB 20 DA 2D  : E6
1FD8  51 EE FF 0A 94 27 02 CF  : D4
1FE0  9C 0A 8D 91 A7 7F 18 02  : 04
1FE8  FA 23 02 68 B5 47 04 E7  : 6E
1FF0  96 7C 29 AD 59 E0 A7 36  : FE
1FF8  1E F1 06 3C AA 20 0B 88  : AE
------------------------------------
SUM:  3F 46 F4 C5 74 46 75 B7  B7E1

2000  51 C0 A6 84 E0 5B 91 6B  : 72
2008  12 D0 D3 AB 8A F3 39 A7  : BD
2010  42 DF EA 1E 70 8A 23 DE  : 24
2018  C8 5B E6 C5 7A 4C D1 A7  : 0C
2020  54 15 E6 87 BD 88 AF EC  : B6
2028  5A 1A 73 70 2B B3 B4 33  : 1C
2030  7D E8 AE D0 7B BF 21 1D  : 5B
2038  A1 0D 7B D1 1E DD E8 AE  : 8B
2040  D4 48 E3 31 AD 87 AE C4  : D6
2048  57 26 65 76 E2 6B B9 66  : C4
2050  F4 22 BE 50 AF 46 2D F5  : 3B
2058  6A 1A 74 C9 7D 74 BC 22  : 90
2060  40 9B D3 25 D6 BF EB 5D  : B0
2068  B7 32 C4 E6 18 AD 55 AB  : 58
2070  77 02 49 8A AE 32 02 09  : 37
2078  FC 5A AB FF 37 38 F9 C8  : 30
------------------------------------
SUM:  2C C1 D0 FE 63 7D B5 9B  A89A

2080  B5 44 EA B8 FB 63 13 F9  : 05
2088  8C D8 A6 9B 80 AC 01 DD  : AF
2090  74 43 73 B8 0E 3C 99 BC  : 81
2098  FE 18 B6 69 9B CF 0A F1  : 9A
20A0  82 B7 3F A8 17 DB EB 92  : 8F
20A8  AF 0E FA B8 83 83 0D BA  : 3C
20B0  B7 9A EA EF 6C F8 32 FE  : BE
20B8  F4 16 7B 94 A1 81 4E 3D  : C6
20C0  4D 9F F5 80 67 88 03 17  : 6A
20C8  8F C3 37 F8 01 64 4A DB  : 0B
20D0  5C 33 7B DF 20 94 E4 58  : D9
20D8  7B 68 76 C2 0E E9 B0 56  : 18
20E0  51 AC 9F 07 7D 75 EA 19  : 98
20E8  9C 84 79 44 DE A8 F5 DF  : 37
20F0  59 38 1E F4 63 64 F5 6A  : C9
20F8  D2 27 86 2D C6 8B E5 A8  : 8A
------------------------------------
SUM:  5A 78 30 DC E5 66 C9 B4  BD9E

2100  87 77 D3 AB CD 47 AE 2E  : 6C
2108  FD 5B E4 3F AB 92 C1 86  : FF
2110  FC 75 04 95 78 E9 FD 3A  : A2
2118  0F 29 BA 21 F9 06 C4 03  : D9
2120  7C 4B 0C 17 AA AD F3 ED  : 21
2128  83 95 E0 33 19 85 9C 3C  : A1
2130  BD 36 34 0D F8 2F 2E C5  : 4E
2138  3F EA 06 0F 52 6B 40 19  : 54
2140  FA 8B 38 1E 39 47 97 62  : 54
2148  9C 64 89 51 A5 67 1C 8F  : 91
2150  1C 6D 5C A7 97 6C 5E 9A  : 87
2158  CF 80 34 3B CB C4 E2 06  : 35
2160  9A C2 67 5A F5 8B 12 65  : 14
2168  A8 D1 FD 10 E8 68 DD E2  : 95
2170  C8 C8 95 65 25 E5 D8 43  : AF
2178  FF 3A CB B5 2C 0D 5C A2  : F0
------------------------------------
SUM:  14 E1 B0 DB 64 57 43 B5  139C

2180  E0 B1 7F 08 FB AD C0 7C  : FC
2188  75 F8 F5 A7 15 D0 48 89  : BF
2190  9D 76 57 32 21 AD 3C 32  : D8
2198  3C D7 E6 B2 C7 BC BB 91  : 7A
21A0  10 31 7A 33 25 97 9B 93  : D8
21A8  B6 A0 2C 9C 88 79 D7 84  : 7A
21B0  A0 51 87 76 D0 8D A4 65  : 54
21B8  A2 F9 52 D6 B0 89 45 EF  : 30
21C0  E2 C6 EB 15 BD A5 C1 F8  : C3
21C8  B7 32 34 C2 40 DF 03 FA  : FB
21D0  FB 44 79 C4 FE 73 4E 46  : 81
21D8  FF 4E 90 47 FB 6D E6 25  : 97
21E0  D6 E7 5D 2B 3C 4A C3 ED  : 7B
21E8  79 CA 1D 59 79 31 C7 3D  : 67
21F0  10 49 F3 2D D3 AA 98 03  : 91
21F8  6F 90 9D F4 EA AD 1B D4  : 16
------------------------------------
SUM:  97 25 62 35 8D 42 8F 91  F599

2200  02 17 19 FB 58 38 40 E9  : E6
2208  3F 09 2A F7 FD 68 D5 F8  : 9B
2210  85 88 D7 F4 2A A1 BB 3B  : 99
2218  0F 6A 37 8B 78 9E 98 E6  : CF
2220  2B 00 71 0B E3 96 0B B3  : DE
2228  96 0B D9 FB D0 57 EE C5  : 4F
2230  38 C5 1C 8A 47 3A C7 D3  : BE
2238  46 94 B5 39 41 05 ED 7D  : 78
2240  54 BA CD B2 F6 9C ED 04  : 10
2248  18 ED F6 2B D9 F3 B8 0B  : B5
2250  A3 BB 4B AC 0A 75 45 2A  : 43
2258  FC FB 31 E2 13 D3 69 09  : 62
2260  1F A6 DB A2 39 05 4E 43  : 11
2268  F2 89 02 D9 44 85 73 86  : 18
2270  C1 28 F8 4B EB 40 CE B7  : DC
2278  DB 91 00 4C D7 3B 9D B8  : 1F
------------------------------------
SUM:  CC BB 80 B7 5D E7 94 44  5D86

2280  53 28 B9 70 6E A8 26 DD  : BD
2288  14 F4 C8 69 41 29 D2 81  : F6
2290  9C 28 F9 47 71 04 89 D2  : D4
2298  1F E6 94 74 3C FA A5 5E  : 46
22A0  0F 3D 1C 7C A1 C8 75 CB  : 8D
22A8  8C 61 92 65 94 AB CF B8  : AA
22B0  5A DB EE 5A 65 55 34 83  : EE
22B8  80 FC 77 06 4F 4D 1F 4D  : 01
22C0  BC 6C 0F 37 07 6A E4 A1  : 64
22C8  63 A9 3C 29 ED 0B 3A AB  : 4E
22D0  81 E2 96 19 BC 58 9C 4F  : 11
22D8  2A 2C 48 06 57 82 FE 74  : EF
22E0  75 8A 5F 4F 49 C2 1C 9B  : 6F
22E8  AF 89 6C EE 44 33 E9 7D  : 6F
22F0  30 8A E3 F2 47 D3 4F FB  : F3
22F8  6A F3 A4 69 81 6C BF AD  : C3
------------------------------------
SUM:  1F 52 9C EC A1 67 88 B0  2D5A

2300  F7 90 32 55 85 CE F9 09  : 63
2308  D0 1D C4 AB 66 29 A0 5A  : E5
2310  31 F8 B7 FD E7 9E 2F 09  : 9A
2318  62 9D 36 92 CE C9 39 E7  : 7E
2320  48 AE 35 58 56 5D D4 67  : 71
2328  D2 32 8A 52 14 EB FB F8  : D2
2330  06 2E 45 3A 35 5E D9 43  : 62
2338  2B BB 2A 83 E0 5C 9C 13  : 7E
2340  CD F4 EA AC DD 7D F8 B6  : 5F
2348  54 60 53 CE FB F1 63 65  : 89
2350  16 77 65 41 19 F4 6A AB  : 55
2358  67 92 32 8A F2 C5 6E A2  : 7C
2360  1F 0F 64 E0 A1 01 C5 E3  : BC
2368  FE E5 F2 28 87 C5 C7 09  : 19
2370  45 E3 B0 6A EF 10 6E 10  : BF
2378  B6 3F 07 75 25 B7 09 7F  : D5
------------------------------------
SUM:  5B 7E F2 22 3E 14 7B EB  F740

2380  10 3A 02 9E 96 92 32 34  : 78
2388  31 3E 11 09 0E 84 FA 4F  : 64
2390  16 AF 36 34 0A 57 7E 2D  : 3B
2398  CE 42 74 20 0E 12 72 89  : BF
23A0  4A F6 84 A5 1C 96 69 EB  : 6F
23A8  78 66 EA B8 11 37 F1 38  : F1
23B0  B9 6E 04 30 AA BF 7F 3E  : 81
23B8  C9 36 6F 09 B2 61 BA 3F  : 83
23C0  D8 F1 7D 53 4E FF 4F E5  : 1A
23C8  31 B6 9E 22 E6 EA 93 12  : 1C
23D0  D9 74 FE 39 8D D7 A2 22  : AC
23D8  E1 F2 09 B0 E1 F5 D3 51  : 86
23E0  03 42 B0 A1 61 4D 49 5D  : EA
23E8  DC 56 D6 F6 06 A4 CA 47  : B9
23F0  8F 5F 1F B0 0A 22 3C 3C  : 61
23F8  72 87 A2 D3 D9 F6 1A E3  : 3A
------------------------------------
SUM:  0C F4 07 09 31 2A 6F 06  8946

2400  BC 9A E4 CE 20 44 0B 99  : 10
2408  6E 21 55 FD B2 88 D2 52  : 3F
2410  D1 C5 6D 95 EE 89 75 EA  : 6E
2418  E8 85 2E D5 B0 14 CF 2A  : 2D
2420  EC 13 8F 08 BC 42 70 7D  : 81
2428  B0 84 9E 16 76 7C 53 CE  : FB
2430  16 38 51 95 24 2D EF C4  : 38
2438  2B 1D D2 94 84 FA 0E 7D  : B7
2440  23 C3 A5 32 AE 10 11 52  : DE
2448  C5 DE F4 5D F9 10 4B C7  : 0F
2450  7E A5 98 A7 C9 CE ED CF  : B5
2458  EB BB 10 AB 70 4A C7 21  : 03
2460  3A D5 A9 75 62 C7 72 A5  : 6D
2468  58 A6 E5 4E 15 96 93 3B  : AA
2470  AF 3E 0F ED A9 8E B2 D1  : A3
2478  67 74 A7 F3 D8 9F 1E FC  : 06
------------------------------------
SUM:  B9 1F A9 00 22 10 C6 41  7DD8

2480  F0 78 E7 CF 07 D5 02 65  : 61
2488  8F 58 C0 8F 13 D5 9D 2D  : E8
2490  6E 4F 5F 88 7C 0C C8 42  : 36
2498  F3 30 2E 50 DD 99 D0 59  : 40
24A0  2E EC D8 43 E0 FC C4 B1  : 86
24A8  28 DD A3 79 09 80 C3 AF  : 1C
24B0  25 46 F2 7E 36 A0 05 B5  : 6B
24B8  09 9D 47 32 5B 0B 36 F9  : B4
24C0  33 41 74 CF D5 EE 83 64  : 61
24C8  75 DA 1C 9B 61 E7 53 33  : D4
24D0  CC CC F6 B9 7D 71 56 9A  : 25
24D8  C5 20 35 F8 62 CD 3F B8  : 38
24E0  5C CF 0E 48 BB 87 5E 77  : 98
24E8  E6 F2 62 40 AA 7F 70 7B  : 8E
24F0  A6 5C E2 9A C9 57 A6 B1  : F5
24F8  59 C1 A9 26 75 4A DE 4B  : D1
------------------------------------
SUM:  DE E0 9E 05 A5 30 B6 12  B0F1

2500  61 05 2D CC 18 A6 6E CE  : 59
2508  4C 2B AE 81 54 72 77 01  : E4
2510  FE 57 30 6E 0B 02 2D 53  : 80
2518  6A 10 6B 51 3A 75 6D FD  : 4F
2520  8B B4 91 69 AE B5 A9 60  : A5
2528  43 CA 72 59 F8 FC FF 94  : 5F
2530  A4 94 DA 51 1E BE 25 9C  : 00
2538  FE 22 E0 9A AB 22 D1 D5  : 0D
2540  71 BC FF 62 CD BB FD 4C  : 5F
2548  D8 3A 65 95 C9 9B 26 6B  : 01
2550  F8 95 B4 A9 EB D9 ED 61  : FC
2558  75 77 42 43 8D 9B EC 71  : F6
2560  B7 F0 CB 2B 13 33 8F EC  : 5E
2568  44 13 9F F4 33 56 79 22  : 0E
2570  AC 58 9E 1E BD 8E BA 4F  : 14
2578  3B 03 87 F0 23 4E 9E 0E  : D2
------------------------------------
SUM:  1D 2B 1C C9 54 4F 79 78  8423

2580  08 24 58 BC 75 3A 21 93  : A3
2588  1B 88 22 E1 AE 11 06 DD  : 48
2590  6A 11 4B A5 9D A6 BA 27  : 8F
2598  84 9B 0D 85 1D 5F F9 88  : AE
25A0  BC 6D 9B 15 F8 24 35 E6  : 10
25A8  44 C7 72 29 9D 56 4C 66  : 4B
25B0  2D B3 CF E3 0A 74 61 EB  : 5C
25B8  9A 09 0C 13 48 EB 3F 8F  : C3
25C0  28 56 23 B4 C6 62 D7 93  : E7
25C8  A0 65 47 9A 06 61 22 D4  : 43
25D0  98 91 4B 65 9B 92 B2 56  : 0E
25D8  7B 06 20 E4 1C B0 E3 D8  : 0C
25E0  62 06 1D 5F E2 BB BA 95  : D0
25E8  69 95 8C C6 76 9F 3E DC  : 7F
25F0  F7 21 AF 54 ED 66 5E D1  : 9D
25F8  8B BA 58 FA 85 36 64 2B  : E1
------------------------------------
SUM:  00 10 3F 05 11 24 43 E7  2831

2600  CE A4 7B 37 83 22 C5 A3  : 31
2608  D2 EC 31 01 9F CD 06 6B  : CD
2610  A1 BA DC 8E 8E F8 03 80  : CE
2618  28 82 D1 EA 92 94 A7 12  : 44
2620  4F 38 FD C7 7F C2 78 57  : 5B
2628  44 FB 7B 50 F4 E6 C2 E3  : 89
2630  80 2C 60 52 82 DB CE 02  : 8B
2638  8F C3 88 A6 6D 63 FD D8  : 25
2640  1D E1 5A 8B 37 AA D2 9D  : 33
2648  B3 F8 D2 09 C2 39 E4 B3  : 18
2650  83 4C AE 8B E0 E0 09 94  : 65
2658  A2 6B 1B 96 F2 A0 77 83  : 4A
2660  49 D2 08 60 2B A2 22 C1  : 33
2668  A0 C1 9E E2 D3 33 B9 80  : 20
2670  1C 22 C4 8F FA 05 5D 02  : EF
2678  D7 9D 2B 72 D1 64 37 48  : C5
------------------------------------
SUM:  DC D0 43 B7 38 02 1F A6  0F40

2680  86 E6 6C 80 E8 74 87 0D  : 48
2688  CE 2A 94 D1 E8 6F 47 DA  : D5
2690  70 5C EA 1C D1 83 4A D2  : 42
2698  04 E1 CC B5 38 8D EB 4D  : 63
26A0  CA C4 46 FF 12 15 B7 D1  : 82
26A8  8B EA 26 74 1E E4 F6 73  : 7A
26B0  C1 D0 F9 FC 08 C1 A5 19  : 0D
26B8  1C 58 87 29 F5 FB 92 71  : 17
26C0  75 7B D3 50 12 8D CC C9  : 47
26C8  D0 27 3B 67 24 89 96 AD  : 89
26D0  17 E6 FC CB FE 2B B4 A3  : 44
26D8  7E 99 69 B5 7F AA 5A 68  : 20
26E0  94 A2 BA D4 05 2D 2E 64  : 88
26E8  B3 9E 94 B4 F5 1D 1E 5A  : 23
26F0  A9 09 D0 5F B7 9B 2D 5E  : BE
26F8  69 7F E0 15 AC 65 F3 D5  : B6
------------------------------------
SUM:  2D 0C 13 ED 16 DD C3 46  D678
```

▶しまった！　夏期講習を受けていたおかげで，編集部に暑中見舞いを出すのを忘れてた。いままで，欠かさなかったのに（涙）。まあいいや。偏差値も上がったことだし。

安川　実(18)愛知県

```
2700  67 96 AB E3 35 14 45 6A  : 83
2708  81 4B 0B 63 F8 7B 4A B5  : AC
2710  DD 92 74 6C 03 5C E5 5B  : EE
2718  FB 5A 08 28 11 BC D5 DA  : 01
2720  A3 08 CC A5 DE 34 9E F3  : BF
2728  E6 F6 92 59 24 04 C9 76  : 2E
2730  A7 2E 4E 19 4F 84 CD DB  : B7
2738  BE 9D C1 98 27 31 0C 34  : 4C
2740  FC FC FA 38 56 F8 5C 2E  : 02
2748  17 0D 43 43 3E 5F 8C 69  : 3C
2750  FC 2A 24 F4 89 DE 87 E6  : 12
2758  24 9A 54 0B 3D 7F 0C 33  : 18
2760  F2 F0 41 93 E5 DD 36 57  : 05
2768  E1 1A 07 EA AB CA 2D 56  : E4
2770  8F 58 B7 08 FE BE D8 90  : CA
2778  B3 A1 BE C2 B4 77 2A BA  : E3
-----------------------------------
SUM:  F6 66 11 4A 55 24 69 73  3F31

2780  6A 49 B9 E3 B2 AF 7E 4B  : 79
2788  31 4E 4B 30 83 D5 28 EC  : 66
2790  41 21 AD 46 0E 25 21 35  : DE
2798  44 76 BF 43 B2 4F 19 4C  : 22
27A0  74 C9 44 4D 24 80 B9 28  : 53
27A8  FC 3B CD CB 3F 99 96 48  : 85
27B0  8C C7 29 B7 2E 21 27 85  : 2E
27B8  C3 BB 6B 99 1A 60 EE D8  : C2
27C0  6E DE 9C CF 50 72 08 EE  : 6F
27C8  B4 53 87 59 FC 1A 73 22  : 92
27D0  3E 86 9F F0 02 A8 69 CD  : 33
27D8  87 05 3F 16 EB CC 32 2A  : F4
27E0  A6 FC 60 64 4A C6 C4 04  : 3E
27E8  FF 6A 34 0F FB 49 E0 14  : E4
27F0  37 A6 C3 A5 82 C6 E5 4B  : BD
27F8  40 C3 11 19 FA DE D8 D8  : B5
-----------------------------------
SUM:  E2 3F 7E 63 9A 45 BB C7  C251

2800  86 A4 82 C5 B5 1E 3D 00  : 81
2808  5C 85 D6 E0 8C 6B 90 F5  : 13
2810  8E FC 46 FB 08 59 27 B8  : 0B
2818  F2 9C 44 D1 DB 94 95 3E  : E5
2820  CE CC FC 07 1A A5 FA F0  : 46
2828  25 14 2E 33 7E BF DF F8  : AE
2830  FC D7 2C 91 6A F6 D5 3D  : 02
2838  4E 59 87 FA F9 67 F5 FE  : 7B
2840  FF F7 E2 94 13 4C 24 8F  : 7E
2848  F4 90 E6 F5 5C B9 D0 F3  : 37
2850  8E E9 A1 CE 35 5C C9 A5  : E5
2858  F9 85 AB 64 2F 8D 86 D2  : A1
2860  19 C9 66 73 10 D2 F1 4A  : D8
2868  5F D0 85 BB FC 9B F5 9F  : 9A
2870  04 56 BD 28 54 A3 23 4E  : A7
2878  6C 44 C4 49 86 EA C6 66  : 59
-----------------------------------
SUM:  01 F9 3F 90 D8 1F 3E A4  2FBA

2880  74 21 D7 95 31 0F BC 6B  : 68
2888  E8 01 2F 38 B3 B6 4B 82  : 86
2890  29 B2 BD 46 37 76 E1 62  : CE
2898  7D 51 36 6A 1F 03 0D CA  : 67
28A0  E7 49 B3 F2 63 17 49 D2  : 6A
28A8  63 0F 3D 02 23 39 37 E6  : 2A
28B0  17 BA 75 AC 5F C6 44 9C  : F7
28B8  A3 C2 AC F4 EC 60 D4 07  : 2C
28C0  BB D8 8C C2 A5 3B 1E 7D  : 5C
28C8  4C C4 89 4A F0 E0 65 A0  : B8
28D0  C6 0D 0D DC 1D BC 97 2A  : 56
28D8  6F 05 9F AC 57 BB 16 15  : FC
28E0  C8 4C BB 4C 5B FF 15 5C  : E6
28E8  A0 C6 2B 70 D1 38 C2 B2  : 7E
28F0  9B 58 15 10 38 A6 85 57  : D2
28F8  69 22 85 D8 62 84 47 DD  : F2
-----------------------------------
SUM:  AE 33 4B 49 DA A7 60 12  639E

2900  11 1A 5B 34 D4 3C 9E 60  : C8
2908  6B 2F A8 C5 7B 27 FC 71  : 16
2910  5B 62 5B 69 4C 91 71 3B  : 0A
2918  82 C1 E1 3E C1 38 A3 8D  : 8B
2920  69 C5 5B 41 5D A1 8E E0  : 36
2928  C1 96 1E 63 76 1A CB 1A  : 4D
2930  76 F4 8D 9A 15 E6 9A 71  : 97
2938  D5 BB 71 1D 40 7D 6D 37  : 7F
2940  C1 BB 69 C6 C6 38 75 F8  : 16
2948  43 7B C1 F7 AE 1F 7F 60  : 22
2950  A1 5C 0F BA 61 F7 F8 26  : 3C
2958  5B FB 43 EF F4 18 DF 5B  : CE
2960  0A 3E 87 2A 7E B5 50 FB  : 77
2968  DC 0F BD F4 CB F1 F7 EB  : 3A
2970  78 43 EF 9A 3E FF 24 CF  : 74
2978  7F CE 27 F4 FD 41 F7 26  : C3
-----------------------------------
SUM:  AB 61 8C 0D D1 96 3B EF  7906

2980  D7 BD 74 8C 0C 59 FD FC  : F2
2988  0E 9F B9 EC F6 03 EF 9B  : D5
2990  DC 5A E7 6D 6D BB FB 54  : 01
2998  AD 33 F9 FB 6C FB A3 78  : 56
29A0  76 FA FF BA C6 FB 57 BC  : FD
29A8  63 D8 D7 5B 31 E7 B1 CB  : 01
29B0  1F 7F 18 FB F7 9D 1A 5E  : BD
29B8  82 5C 39 75 52 F5 43 DF  : F5
29C0  EF 31 EB CB 56 5A 72 F5  : ED
29C8  72 EB A5 AD 2F 52 C6 D7  : CD
29D0  57 FC 2C 6E D8 FE 16 39  : 12
29D8  02 EF D2 63 F3 8C 5F 46  : 4A
29E0  F7 97 AB FD ED 67 F4 B1  : 2F
29E8  EA 31 E5 CD BD 68 47 A9  : E2
29F0  D2 B1 C8 D6 7D B6 3F 23  : B6
29F8  1B 5D 9F F1 EB F7 1D 01  : 08
-----------------------------------
SUM:  70 73 B9 3F 7D 38 33 F0  069A

2A00  3F 47 37 C4 97 FB 72 F5  : 7A
2A08  32 F4 D2 BB 99 6B CB D4  : 56
2A10  4B D1 81 EF AF 2E 20 BB  : 44
2A18  CC CB B1 D7 5B 7D AB 99  : 3B
2A20  A1 1D 4E EF 55 E5 CB ED  : ED
2A28  86 26 23 1B 5F 0E DF 61  : 97
2A30  E5 80 4E F1 8F 20 5E FA  : AB
2A38  00 1F A2 97 AC 2C 7B 1C  : C7
2A40  BD 6D DE B2 DF A3 95 9B  : 6C
2A48  B4 65 DA CB D9 4B CD CB  : 7A
2A50  D4 31 B5 B9 F7 35 DF CE  : 4C
2A58  35 4F 47 2B B5 FD 19 5D  : 1E
2A60  66 F3 E6 03 CA 96 C1 70  : D3
2A68  2E A7 BE F0 FF 97 A5 6F  : 2D
2A70  33 54 B1 B5 D5 6C 36 7F  : E3
2A78  CA C7 DD 63 93 7F B8 C0  : 5B
-----------------------------------
SUM:  9F C0 82 43 BE 88 39 30  0215

2A80  B4 0E 74 3F 47 25 EB 56  : 22
2A88  FB 52 76 3A 79 9A 16 6F  : 95
2A90  39 FA EC 7A 02 C7 F0 6C  : BE
2A98  37 1A CF C9 D1 DB 77 3E  : 4A
2AA0  E3 1B E6 3E 0B F0 6F 05  : 91
2AA8  8F 5E E7 DD D9 79 9D 1D  : BD
2AB0  07 AD EF F6 3C 82 19 CB  : 3B
2AB8  92 34 3F 4C 53 EA 00 AF  : 3D
2AC0  8F 81 D6 77 F8 9B F1 9F  : 80
2AC8  53 1A 42 FF 93 5F B3 63  : B6
2AD0  D6 D8 E7 98 DA F8 9F 77  : 15
2AD8  57 F2 35 66 C7 9C C7 26  : 34
2AE0  E7 7D AE FC D7 F9 7E 5C  : B8
2AE8  BC B0 EA FD 14 BD 97 4B  : 06
2AF0  82 C8 17 63 2E B9 8D A8  : E0
2AF8  D7 77 8C 7F 20 A4 F9 DD  : F3
-----------------------------------
SUM:  35 9F 0F 68 6B D7 32 D6  D06F

2B00  46 A3 5B 99 97 AA 18 F5  : 2B
2B08  25 EC 98 F5 8E 1F DA 9C  : C1
2B10  87 F0 05 FB 5D 9F AF 73  : 95
2B18  EB EA 6B DA B1 33 6D C9  : 34
2B20  9E 1A 01 B6 27 A5 D9 9E  : B2
2B28  87 F9 68 03 F6 AF 04 1E  : B2
2B30  14 47 D7 63 CD D1 FE D1  : 02
2B38  0A 5F 0F B3 C2 D4 24 EF  : D4
2B40  A9 7A 80 B1 7A 33 2E 85  : B4
2B48  BB 93 81 67 87 F6 81 E5  : 19
2B50  87 9E 16 A5 FE 9C F9 F2  : 65
2B58  B9 74 7B 97 25 09 24 06  : 97
2B60  DD 1F 22 89 EB C7 0B 34  : 98
2B68  0F 6E 58 72 CD BA E9 5F  : 16
2B70  A4 6A 43 DA D4 9E 9A B3  : EA
2B78  D3 E5 1A 11 F4 4C 01 97  : BB
-----------------------------------
SUM:  27 1D 1B 6C 83 CD 68 88  660B

2B80  A0 32 67 A7 D1 16 34 22  : 1D
2B88  B7 A7 05 BC 38 B5 D1 C2  : 9F
2B90  28 8F 57 C3 EA 26 A3 3E  : C2
2B98  C7 74 6C 89 EC CB 7B BC  : 1E
2BA0  3E C3 76 7F 37 67 D1 6E  : D3
2BA8  8F BE 51 99 B2 4B CE 80  : 82
2BB0  05 D1 1A 15 E9 A7 C7 F6  : 52
2BB8  B5 6B 5B 7D 01 95 DB 8C  : F5
2BC0  AE 44 78 D8 8C 6C 01 A6  : E1
2BC8  F2 A6 B8 36 CC FA DE A0  : CA
2BD0  58 E3 CF 6A E8 01 7E 7C  : 57
2BD8  CB AB F1 6B E9 9C 3E 42  : D7
2BE0  F7 E3 26 89 F9 6E 7C F6  : 62
2BE8  BE 20 C4 D1 FC D9 B4 8F  : 8B
2BF0  B6 F2 07 A5 03 76 A2 C6  : 35
2BF8  A4 01 60 1F 97 5A 0C 96  : B7
-----------------------------------
SUM:  9F 07 AC 5A 6A C4 DD 33  4F36

2C00  16 8A 60 8C BD 6E 40 56  : 4D
2C08  10 09 BD 3E 44 BA 56 90  : F8
2C10  7D 4C 02 61 1F B1 68 05  : 69
2C18  DE 44 0E DF 98 3B 06 6C  : 54
2C20  C0 D9 E5 89 79 33 A6 DE  : 37
2C28  8E 03 74 6E EC 61 F1 E7  : 98
2C30  07 F6 A6 FB 22 BA 61 87  : 62
2C38  2B FC A9 A4 9B 68 3B B6  : 68
2C40  AC D1 46 FD 80 71 0B B0  : 6C
2C48  05 76 35 CC F9 F0 F4 C2  : 1B
2C50  F6 80 5A C5 3E 75 F8 C1  : 01
2C58  D0 1F 8B 4B 06 91 F8 6E  : C2
2C60  78 D9 90 1D 8C 1D 7A A3  : C4
2C68  15 E6 8F D5 32 47 DC B9  : 6D
2C70  F7 8C F8 18 B5 5D A8 04  : 51
2C78  A1 33 60 CF 8B 3F 10 FB  : D8
-----------------------------------
SUM:  9D 55 AC 52 95 31 34 55  2AF4

2C80  A5 E0 D5 EE 4E 55 EE A6  : 7F
2C88  15 2B 93 96 CD 11 FC 5E  : A1
2C90  1F 95 DA FA E4 F9 56 A9  : 64
2C98  78 72 BD 6B 95 5F 9F 9F  : 44
2CA0  69 CE 27 72 7D 83 3D 7E  : 8B
2CA8  58 E5 B2 E7 A6 73 DA 27  : F0
2CB0  0E BF 32 E7 E7 B7 87 26  : 31
2CB8  F7 26 CE 0D 6F A4 3E 06  : 4F
2CC0  88 F6 73 7A A6 01 CC 4E  : 2C
2CC8  E0 75 4B E1 B5 E2 82 13  : AD
2CD0  40 DE D8 C1 A0 7D E6 E4  : 9E
2CD8  74 1D 28 B9 76 06 F3 E0  : C1
2CE0  33 3E 30 39 11 91 A3 39  : 58
2CE8  7A 9F 5E B8 D2 CC D1 A5  : 43
2CF0  D1 34 26 D0 1C DD E9 C0  : 9D
2CF8  1E 78 71 59 74 AB 91 8B  : 9B
-----------------------------------
SUM:  CF 99 BB 25 F1 5A D0 6B  3E2D

2D00  44 2D 6D 16 F2 47 43 E9  : 59
2D08  5B 09 7D 66 61 B5 38 BD  : 52
2D10  A4 39 89 F0 06 AB A4 1D  : C8
2D18  CF DD 01 39 6D 32 32 3A  : F1
2D20  21 66 90 0A E9 0D 5A 3B  : AC
2D28  F6 9C D6 07 21 7A 03 99  : A6
2D30  2B E1 DC 31 C6 95 78 04  : F0
2D38  75 23 5B E6 06 0C 17 FB  : FD
2D40  68 C0 43 E4 0B BA D3 6F  : 56
2D48  07 4B B8 FF DF F0 00 00  : D8
2D50  00 00 00 00 00 00 00 00  : 00
2D58  00 00 00 00 00 00 00 00  : 00
2D60  00 00 00 00 00 00 00 00  : 00
2D68  00 00 00 00 00 00 00 00  : 00
2D70  00 00 00 00 00 00 00 00  : 00
2D78  00 00 00 00 00 00 00 00  : 00
-----------------------------------
SUM:  38 5D 0C B0 86 AB 10 3F  91A8
```

リスト2 ZPP.C

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  メインルーチン  */

#include "Zpp.h"

/*変数宣言*/

struct {
    char *name;
    short s_point;
    short e_point;
    short time;
    short flag;
} block[BLOCKMAX+1],*b_sort[BLOCKMAX+1];
char *master_buf[FILEMAX];
char *macro_buf[26];
char line_buf[LINEMAX];
short fsize=-1,fpoint=-1,b_No_max=-1,b_sor_max=0;
short b_fpoint;
short b_No;              /*パス2での作業中の行番号*/
short work_track=1;      /*作業中のトラックナンバー*/
short debug1=1,debug2=2,debug_time=0;    /*デバッグコマンド用変数*/
short por_mod=0;         /*ポルタメントモード*/
short tie_b,tie_p;       /*ポルタメント変換用変数*/
char key_s[7];           /*キースケール*/
char m_key_s[7]={0,2,4,5,7,9,11};
                         /*Cメジャーのキースケール*/
char ks_type[16][7];/*キースケールのバッファ*/
char key_str[12][4]=
    {"c","d-","d","e-","e","f","g-","g","a-","a","b-","b"};
char oct,key_no,out_oct,tie,tie_key;
char ext[4],buffer[LINEMAX];
char modflag=1,b_set_mod=0;
FILE *fpi,*fpo,*zms;

/*呼び出し関数宣言*/

extern void block_search();
extern void command_search();

/*関数定義*/

static void help(){
    printf("Zpp.x  Z-muSiCプリプロセッサVer1.00");
    printf("¥tCopyright 3/13 1992 by M.Okamoto¥n");
    puts("使用法:Zpp [スイッチ] [入力ファイル名] [出力ファイル名]");
    puts("¥t/d¥t出力ファイルにもとの行番号もつける");
    puts("¥n¥t入出力ファイル名の拡張子を指定しなければそれぞれZPS、ZMSとする");
    puts("¥t出力ファイル名を指定しなければ入力ファイル名の拡張子を ZMSとし");
    puts("¥tたものを出力ファイル名とする");
    exit(1);
}

static char *set_filename(char *i_fname,char *i_ext){
    if (stcgfe(ext,i_fname)==0){
        strmfe(buffer,i_fname,i_ext);return(buffer);}
    strcpy(buffer,i_fname);
    return(buffer);
```

▶ 8月17日の朝，会社にて。同僚「すごい日焼けだね。どこで焼いたの？」。私「晴海」。
山田　俊英(24)東京都

```
}

static void err_exit(){
   perror(buffer);
   exit(-1);
}

static void loadfile(){
   while (fgets(buffer,LINEMAX,fpi)){
      if (++fsize > FILEMAX){
         puts("行数が多過ぎます");exit(-1);}
      *(buffer +strcspn(buffer,"/¥n")) = NULL;
      if ((master_buf[fsize]=strdup(buffer)) == NULL){
         puts("メモリが足りません");exit(-1);}
   }
   fsize++;
}

static void macro_init(){
   char i;
   char str[2]="a";
   for (i=0;i<26;i++){
      str[0]=i+0x61;
      if ((macro_buf[i]=strdup(str)) == NULL){
         puts("メモリが足りません");exit(-1);}
   }
}

void main(int argc,char *argv[]){
   char flag=1;
   if (argc==1) help();
   if ((*argv[1]=='-')||(*argv[1]=='/')){
      flag=2;
      switch (*(argv[1]+1)){
         case 'h':
         case 'H': help();
         case 'd':
         case 'D': modflag=2;break;
      }
   }
      /*ファイルオープン*/
   if (argc==flag) {puts("入力ファイルを指定してください");exit(-1);}
   if ((fpi=fopen(set_filename(argv[flag],"ZPP"),"r")) == NULL) err_exit();
   if (++flag==argc) strmfe(buffer,argv[flag-1],"");
   else strncpy(buffer,argv[flag],255);
   if ((fpo=fopen(set_filename(buffer,"ZMS"),"w")) == NULL) err_exit();

   allmem();
   macro_init();
   loadfile();/*ここでfsize-1 の値のところまでバッファを使用している*/

      /*メインルーチン*/
   block_search();
   command_search();

   end();
}
```

リスト3 ZPP1.C

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  ブロック割り当てルーチン  */

#include "Zpp_HED.h"
#include "Zpp_SUB.h"

/*関数定義*/

static void name_cheak( char *work ){
   short block_No;
   int ret;
   if ( (ret=block_match(work,&block_No))==0 ){
      error_puts(fpoint,"同じブロック名があります");return;}
   if (ret>0) ++block_No;
   {
      short i;
      for ( i=++b_sor_max ; i > block_No ; --i )
         b_sort[i]=b_sort[i-1];
   }
   b_sort[block_No]=&block[b_No_max];
}

static void block_check(){
   char *work;
   char *b_name_epoint;
   work = master_buf[fpoint]+1;
   if (*work == B_HEAD ) {
      error_puts(fpoint,"ブロック中でないのにブロック終了宣言があります");
      master_buf[fpoint]=master_buf[fpoint]+2;}
   else {
      if ((b_name_epoint=strchr(work,B_HEAD)) == NULL )
         error_puts(fpoint,"ブロック名を*で括っていません");
      else {
         if (++b_No_max == BLOCKMAX){
            error_puts(fpoint,"ブロックバッファが溢れました");
            end();}
         *b_name_epoint=NULL;
         master_buf[fpoint]=b_name_epoint+1;
         block[b_No_max].name = work;
         block[b_No_max].s_point=fpoint;
         name_cheak(work);
         b_set_mod=1;
      }
   }
}

static void block_end(short endpoint){
   block[b_No_max].e_point=fpoint-1;
   b_set_mod=0;
   if ( ! endpoint ) {
      if ( *(master_buf[fpoint]+1) != B_HEAD )
         block_check();
      else master_buf[fpoint]=master_buf[fpoint]+2;
   }
}

static void block_set(char flag){
   if (b_set_mod){
      if ( !flag ) block_end(0);
      else {
         error_puts(fpoint-1,"ブロック終了宣言がありません");
         block_end(-1);
      }
   }
   else if ( !flag ) block_check();
}

static void pass1(){
   while (++fpoint<fsize){
      if ( ! strncmp(master_buf[fpoint],EXT_ON) ) return;}
}

void block_search(){
   pass1();
   while (++fpoint<fsize){
      if ( ! strncmp(master_buf[fpoint],EXT_OFF) ){
         block_set(-1);pass1();continue;}
   if ( *master_buf[fpoint] == B_HEAD ) block_set(0);
   }
   block_set(-1);++b_No_max;
/*
   {
      short i;
      for ( i=0 ; i<b_No_max ; ++i)
         printf("%5d => %5d :%s¥n",block[i].s_point,block[i].e_point,block[i].name);
   }
   {
      short i;
      for ( i=0 ; i<b_sor_max ; ++i){
         printf(" %3d¥t%s¥n",i,b_sort[i]->name);}
   }
*/
}
```

リスト4 ZPP2.C

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  パス2・レベル1:コマンド分岐 */

#include "Zpp_HED.h"
#include "Zpp_SUB.h"

/*関数定義*/

extern char block_call(short block_No);

void block_reset(){
   register int i;
   for ( i=0 ; i<BLOCKMAX ; i++ )
      block[i].time=0;
   debug_time=0;
   out_oct=oct=4;key_no=60;
   tie=tie_b=tie_p=0;tie_key=60;
   ks_set(0,0,MAJ);
}

static void ks_init(char x,char a){
   static char type[3][7]= {{2,2,1,2,2,2,1}
                           ,{2,1,2,2,1,2,2}
                           ,{2,1,2,2,1,3,1}};
   char i;                   /* c d e f g a b<c */
   for (i=0;i<7;i++)ks_type[x][i]=type[a][i];
}

static char *macro_set(char *str){
   char m_no;
   ++str;br_skip(str);
   if ( ((m_no=*str)<'A')||(m_no>'Z')){
      error_puts(fpoint,"マクロ定義できる文字ではありません");return(str);}
   m_no-='A';
   ++str;br_skip(str);
   if ( *str==',' ){
      ++str;br_skip(str);
      if ( *str=='"' ){
         macro_buf[m_no]=++str;
         str_skip2(str,'"',')');
         if ( *str=='"' ){
            char *buf=macro_buf[m_no];
            *str=NULL;
            while (buf!=str){
               *buf=(*buf>='A'&&*buf<='Z')?(*buf-('A'-'a')):*buf;
               ++buf;
            }
            return(++str);
         }
      }
   }
   error_puts(fpoint,"文法エラーです");
   return(str);
}

static char *track_set(char *str){
   short t=work_track;
   ++str;
   if ( (work_track=figure_set(&str))!=t ) block_reset();
   return(str);
}

static char *debug_set(char *str){
   ++str;
   if ( (debug1=figure_set(&str))==0 ){
      debug2=0;return(str);}
   br_skip(str);
   if ( *str==',' ){
      ++str;debug2=figure_set(&str);}
   else debug2=debug1+1;
   return(str);
}

static char *por_set(char *str){
```


▶今年の夏，初めて金縛りというものを体験しました。体は眠っているけど頭だけが起きていました。こういう現象だとわかってはいましたが，実際に体が動かないと一瞬どきっとするし，あまり気分のいいものではないですね。　中村　朋弘(18)愛知県

```
    int a;
    ++str;
    if ( (a=figure_set(&str))>=0 && a<=1 ) por_mod=a;
    return(str);
}/*モード2はGS規格モードとして予約します*/

static char *key_scale_set(char *str){
    char x;
    ++str;br_skip(str);
    if (*str<'0'||*str>'9'){
        error_puts(fpoint,"音階が登録できません");return(str);}
    x=*str-'0';ks_init(x,0);
    ++str;str_skip2(str,',',')');
    if (*str==NULL) return(str);
    while ( *str!=')' || *str!=NULL ){
        char key,half=0;
        ++str;br_skip(str);key=*str;
        if (key>='A'&&key<='G') key=key-('A'-'a');
        if (key<'a'||key>'g') return(str);
        key-='a';
        while(1){
            if (*(++str)=='-'){
                --half;continue;}
            if ((*str=='+')||(*str=='#')){
                ++half;continue;}
            break;
        }
        ks_type[x][(key-3+7)%7]+=half;
        ks_type[x][(key-2+7)%7]-=half;
        --str;
    }
    return(str);
}

static void pass2(){
    while (++fpoint<fsize){
        if ( ! strncmp(master_buf[fpoint],EXT_ON) ) return;
        output(master_buf[fpoint]);
    }
}

void command_search(){
    short b_pass_No=0;
    char top[]="top level",tflag=0;
    fpoint=-1;

    {
        int i;
        for (i=0;i<11;++i) ks_init(i,0);
    }
    ks_init(11,1);/*        短音階*/
    ks_init(12,2);/*和声的短音階*/

    b_sort[BLOCKMAX]=&block[BLOCKMAX];
    block[BLOCKMAX].name=top;
    block_reset();

    pass2();
    while (++fpoint<fsize){
        char *str;str=master_buf[fpoint];
        if ( fpoint == block[b_pass_No].s_point ){
            fpoint=block[b_pass_No].e_point;
            ++b_pass_No;
            continue;
        }
        if ( ! strncmp(str,EXT_OFF) ){
            pass2();continue;}
        br_skip(str);
        if (tflag==0) switch (*str){
            case'(':++str;br_skip(str);
                    switch (*str){
                        case'#':str=macro_set(str);break;
                        case't':
                        case'T':str=track_set(str);
                                tflag=1;break;
                        case'd':
                        case'D':str=debug_set(str);break;
                        case'@':++str;br_skip(str);
                                if (*str=='p'||*str=='P')
                                    str=por_set(str);
                                if (*str=='k'||*str=='K')
                                    str=key_scale_set(str);
                    }
                    str_skip(str,')');
                    if (*str == NULL)continue;
                    master_buf[fpoint]=str+1;fpoint--;
            case 0 :continue;
        }
        tflag=0;
        master_buf[fpoint]=str;
        block[BLOCKMAX].s_point=block[BLOCKMAX].e_point=fpoint;
        block[BLOCKMAX].time=block[BLOCKMAX].flag=0;
        block_call(BLOCKMAX);
    }
}
```

リスト5 ZPP3.C

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  パス2:ブロック呼び出し、特殊展開   */

#include "Zpp_HED.h"
#include "Zpp_SUB.h"

/*関数定義*/

extern void macro_call(char *s_str,char *e_str);

static void No_block(){
    if (buffer[0]==':'&&buffer[1]==NULL)
        if (debug1!=(++debug_time)&&debug2!=debug_time)
            return;
    if ((buffer[0]=='p'||buffer[0]=='P')&&buffer[2]==0)
        switch (buffer[1]){
            case'0':por_mod=0;return;
            case'1':por_mod=1;return;
/*          case'2':por_mod=2;return; */
        }
    sprintf(line_buf,"[%s]",buffer);
    track_out(line_buf);
}

char block_call(short block_No){
    static short level=-1;
    short i = b_sort[block_No]->s_point;
    ++level;
    ++b_sort[block_No]->time;
    if ( (b_sort[block_No]->flag & 1)==1 ){
        --level;return(-1);}
    b_sort[block_No]->flag|=1;

    if (modflag==2){
        sprintf(buffer,"¥n/ level %d: time=%d BLOCK [%s] start¥n"
        ,level,b_sort[block_No]->time,b_sort[block_No]->name);
        output(buffer);
    }

    for ( ; i<=b_sort[block_No]->e_point ; i++ ){
        char b_pass='[',pass2=0,pass3=0,*str,*s_str;
        str=master_buf[i];
        b_fpoint=i;
        while(1){
            if (*str==NULL) break;
            s_str=str;
            str_skip2(str,b_pass,'"');
            if ((b_pass!=0)&&(s_str!=str))
                macro_call(s_str,str);
            if (*str==NULL) break;
            if (*str=='['){              /*ブロック呼び出し処理*/
                short call_B_No;
                ++str;
                {
                    char *c_str;
                    c_str=buffer;
                    while((*str!=0)&&(*str!=']')){
                        *c_str=*str;
                        c_str++;str++;
                    }
                    *c_str=NULL;
                }
                if (*str==NULL){
                    B_err("[]の対応が異常です");continue;}
                ++str;
                if (block_match(buffer,&call_B_No)){
                    No_block();continue;}
                if (block_call(call_B_No))
                    error_puts(i,"ブロックの再帰呼び出しはできません");
                continue;
            }
            if ((pass2^=1)==1) {    /*特殊展開コマンド処理*/
                b_pass=0;++str;
                if (pass3==0) {
                    str--;
                    do{
                        int x,y;
                        ++str;x=figure_set(&str);br_skip(str);
                        if (*str != ','){
                            if (x==b_sort[block_No]->time||x==0){
                                b_pass='[';pass3=1;}
                            break;}
                        ++str;y=figure_set(&str);br_skip(str);
                        if ((x==b_sort[block_No]->time||x==0)&&
                            (y==work_track||y==0)){
                                b_pass='[';pass3=1;break;}
                        br_skip(str);
                    }while(*str == ',');
                }
                while(*str == ','){
                    ++str;figure_set(&str);br_skip(str);}
            }else{
                b_pass='[';
                if ( *(++str)!='"' ) pass3=0;
            }
        }
    }
    if (modflag==2){
        sprintf(buffer,"¥n/ [%s] end",b_sort[block_No]->name);
        output(buffer);
    }
    b_sort[block_No]->flag&=0xfffe;
    level--;
    return(0);
}
```

リスト6 ZPP4.C

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  パス2:マクロ展開  */

#include "Zpp_HED.h"
#include "Zpp_SUB.h"

/*関数定義*/

void macro_call(char *str,char *e_str){
    char *line=line_buf;
    for (;str!=e_str;++str){
        if (*str=='@'){         /*@コマンドパス*/
            *line='@';++line;
            if ((++str)==e_str) break;
            *line=((*str>='A')&&(*str<='Z'))?
                (*str-('A'-'a')) : *str;
            ++line;continue;
        }                       /*マクロセット*/
        if ( (*str>='A')&&(*str<='Z') ){
            char *copy;
            copy=macro_buf[*str-'A'];
            while (*copy!=NULL){
                *line=*copy;
                ++line;++copy;
            }
            continue;
        }
        *line=*str;++line;
    }
    *line=0;
    key_scale_change(line_buf);
}
```

▶ 9月号の特集はよかったです。「最小2乗法」は，ちょうど学校で習っているあたりなので，とても参考になりました。今度はカオスやフラクタルなんかについてやってもらえるとうれしいですね。
福島　敦(20)神奈川県

リスト7　ZPP5.C

```c
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  パス2：キースケール変更  */
/*  ここで音程を128～255の絶対値に変換する  */
#include "Zpp_HED.h"
#include "Zpp_SUB.h"
/*関数定義*/
extern void ext_portamento (char *str);
void key_scale_change (char *str){
    char *out,*outbuf,octx,octf=0;
    outbuf=( (str==buffer) ? line_buf : buffer );
    out=outbuf-1;
    br_skip(str);
    str--;
    while(*(++str)!=NULL){
                /*ブランクパス*/
        if (*str==' '||*str=='¥t') continue;
                /*@コマンドパス*/
        if (*str=='@'){
            *(++out)='@';
            if (*(++str)==0) break;
            *(++out)=*str;continue;
        }
                /*音階セット*/
        if (*str>='a'&&*str<='g'){
            short key;
            if (*(str+1)=='%'){
                key=12*(oct+1)+m_key_s[(*str-'c'+7)%7];
                ++str;
            }
            else key=12*(oct+1)+key_s[*str-'a'];
            while(1){
                if (*(++str)=='-'){
                    --key;continue;}
                if ((*str=='+')||(*str=='#')){
                    ++key;continue;}
                --str;break;
            }
            if (key<0){
                key=0;B_err("音階が低過ぎます");}
            else if (key>127){
                key=127;B_err("音階が高過ぎます");}
            key_no=key;*(++out)=key|128;
            continue;
        }
                /*オクターブチェンジ*/
        if (*str=='<') {++oct;continue;}
        if (*str=='>') {--oct;continue;}
        if (*str=='o'){
            ++str;
            oct=figure_set(&str);
            --str;continue;
        }
                /*和音コマンドのオクターブ補正*/
        if (*str=='¥''){
            *(++out)='¥'';
            (octf=~octf) ? (octx=oct) : (oct=octx);
            continue;
        }
                /*キースケールチェンジ*/
        if (*str=='k'&&*(str+1)=='s'){
            char key=0,half=0,type=10;
            if (*(str+=2)>='a'&&*str<='g'){
                key=(*str-'c'+7)%7;
                while(1){
                    if (*(++str)=='-'){
                        --half;continue;}
                    if (*str=='+'||*str=='#'){
                        ++half;continue;}
                    break;
                }
            }
            if (*str>='0'&&*str<='9'){
                type=*str-'0';++str;}
            if ((*str=='m')&&((*(str+1)<'0')||(*(str+1)>'9'))){
                if (type==10) type=0;
                type+=11;++str;
            }
            ks_set(key,half,type);--str;continue;
        }
        *(++out)=*str;
    }
    *(++out)=NULL;
    ext_portamento (outbuf);
}
```

リスト8　ZPP6.C

```c
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  パス2：拡張ポルタメントコマンド  */
#include "Zpp_HED.h"
#include "Zpp_SUB.h"
/*関数定義*/
static void B_set(char **out,char key){
    int pitch=8192*(key-tie_key)/12+tie_b;
    if      (pitch>=8192) pitch= 8191;
    else if (pitch<-8192) pitch=-8192;
    if (tie_p!=pitch){
        *out=*out+sprintf(*out+1,"@b%d",pitch);
        tie_p=pitch;
    }
    *(++*out)=tie_key|128;
}
static char P_set(char **string,char **output){
    char *str=*string,*out=*output;
    char key1,key2,lf=0,pd=0;/*lf=0:ｿｳﾀｲｵﾝﾁｮｳ lf=1:ｾﾞｯﾀｲｵﾝﾁｮｳ */
    unsigned short le=65535,de=65535;
    int pitch1,pitch2;
    if (por_mod==0) {*(++*output)='(';return(0);}
    if (!(*(++str)&128)) {*string=str;return(-1);}
    key1=*str&127;
    if ((*(++str)>='0')&&(*str<='9')) le=figure_set(&str);
    while (*str=='.') {++pd;++str;}
    if (!(*str&128)){*string=str;return(-1);}
    key2=*str&127;
    ++str;str_skip(str,')');
    if (*str==0) {*string=str;return(-1);}
    if ((*(++str)>='0')&&(*str<='9')) {le=figure_set(&str);lf=1;}
    if (*str==','){
        ++str;de=figure_set(&str);}
    if (tie!=2) tie_key=key1;
    pitch1=8192*(key1-tie_key)/12+tie_b;
    if      (pitch1>=8192) pitch1= 8191;
    else if (pitch1<-8192) pitch1=-8192;
    pitch2=8192*(key2-tie_key)/12+tie_b;
    if      (pitch2>=8192) pitch2= 8191;
    else if (pitch2<-8192) pitch2=-8192;
    out=out+sprintf(out+1,"@b%d,%d",pitch1,pitch2);
    if (de!=65535) out=out+sprintf(out+1,",%d",de);
    *(++out)=tie_key|128;
    if (le!=65535){
        if (lf==1) *(++out)='*';
        out=out+sprintf(out+1,"%d",le);
    }
    while ((--pd)!=-1) *(++out)='.';
    tie=1;
    tie_p=9999;
    *output=out;*string=str-1;
    return(0);
}
void ext_portamento (char *str){
    char *out,*outbuf;
    outbuf=( (str==buffer) ? line_buf : buffer );
    out=outbuf-1;
    str--;
    while(*(++str)!=NULL){
        if (*str&128){
            char key=*str&127;
            if (tie==0){
                tie_key=key;*(++out)=*str;continue;}
            if ((--tie)==0) tie_key=key;
            if (tie==2){
                *(++out)=*str;tie=1;continue;}
            if (por_mod!=0) B_set(&out,key);
            else                *(++out)=*str;
            continue;
        }
        if (*str=='&'){ tie=2;*(++out)='&';continue;}
        if (*str=='@'){
            if (*(str+1)=='b'){
                char *s=(str+=2)-1;
                tie_b=figure_set(&str);
                *(++out)='@';*(++out)='b';
                s_cpy(out,s,str);
                tie_p=tie_b;
                if (*str!=',') {--str;continue;}
                s=str++;
                tie_b=figure_set(&str);
                *(++out)=',';
                s_cpy(out,s,str);--str;
                tie=3;
                continue;
            }
            if (*(str+1)=='k'){
                char *s=(str+=2)-1;
                tie_b=figure_set(&str)*8192/768;
                *(++out)='@';*(++out)='k';
                s_cpy(out,s,str);--str;
                continue;
            }
            *(++out)=*str;continue;
        }
        if (*str=='('){
            if (P_set(&str,&out))
                B_err("ポルタメントコマンドが異常です");
            continue;
        }
        if (*str=='|'){
            if (*(++str)==':'){
                *(++out)='|';*(++out)=':';continue;}
            if ((tie!=0)&&(tie_p!=tie_b)){
                out=out+sprintf(out+1,"@b%d",tie_p=tie_b);}
            *(++out)='|';--str;continue;
        }
        if (*str==':'&&*(str+1)=='|'){
            if ((tie!=0)&&(tie_p!=tie_b)){
                out=out+sprintf(out+1,"@b%d",tie_p=tie_b);}
            *(++out)=':';*(++out)='|';
            ++str;continue;}
        *(++out)=*str;
    }
    *(++out)=NULL;
    MML_track_output(outbuf);
}
```

リスト9　ZPP_OUT.C

```c
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  MMLデータ出力  */
#include "ZPP_hed.h"
/*関数定義*/
void MML_track_output(char *str){
    char *out,*outbuf,octx,octf=0;
    if (*str==0) return;
    outbuf=( (str==buffer) ? line_buf : buffer );
    str--;
```

▶そうかそうか，V70には小人さんが住んでいるのか。「小人さんの欠点は計算を間違えることぐらい」という格言を思い出す今日この頃でした。　大庭　徹也(22)三重県

```
    out=outbuf+sprintf(outbuf,"(t%d)¥t",work_track)-1;
    while(*(++str)!=NULL){
       if (*str&128){
          char key,*keystr;
          key=(*str&127)-((out_oct+1)*12);
          while(key>11){
             key-=12;*(++out)='<';++out_oct;}
          while(key<0){
             key+=12;*(++out)='>';--out_oct;}
          keystr=key_str[key]-1;
          while(*(++keystr)!=0) *(++out)=*keystr;
          continue;
       }
       if (*str=='¥''){
          *(++out)='¥'';
          (octf^=1) ? (octx=out_oct) : (out_oct=octx);
          continue;
       }
       *(++out)=*str;
    }
    if (modflag==2)
       sprintf(++out," /%5d",b_fpoint+1);
    else *(++out)=NULL;
    output(outbuf);
 }
```

リスト10 ZPP_SUB.C

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  共通サブルーチン  */
#include "ZPP_hed.h"
/*関数定義*/
void error_puts(short err_line,char *err_mes){
   printf("%s %5d : %s¥n",fpi->_fname,err_line+1,err_mes);
   modflag&=2;
}
void end(){
   if (modflag!=0){
      exit(0);}
   strcpy(buffer,fpo->_fname);
   fcloseall();
   remove(buffer);
   exit(-1);
}
void output(char *outstr){
   if (modflag != 0) {
      fputs(outstr,fpo);
      if (modflag == 2)
         fprintf(fpo," /%5d",fpoint+1);
      fputs("¥n",fpo);
   }
}
void track_out(char *buf){
   char *buf2;
   buf2=( (buf==buffer) ? line_buf : buffer );
   if (modflag!=2) sprintf(buf2,"(t%d)¥t%s",work_track,buf);
   else sprintf(buf2,"(t%d)¥t%s /%5d",work_track,buf,b_fpoint+1);
   output(buf2);
}
static char block_match_sub(char *name,short *width,short *look){
   if (*look<b_sor_max){
      int ret;
      if ( (ret=strcmp(name,b_sort[*look]->name))==0 ) return(0);
      if (ret>0) { *look=*look+*width;return(1);}
   }
   *look=*look-*width;return(-1);
}
/* char *name が b_sort にあるかを調べる
   b_No にその名前のブロック番号に一番近いものを返す
   return は b_no が完全に同じかどうかのフラグ
   1 ならnameの方が大きく、-1なら小さい
*/
short block_match ( char *name , short *block_No) {
   if (b_sor_max == 0){*block_No=-1;return(1);}
   {
      short width=BLOCKMAX>>2,look=BLOCKMAX>>1;
      char ret;
      do{
         if ( block_match_sub(name,&width,&look) == 0 ){
            *block_No=look;return(0);}
      }while ( (width>>=1) != 0 );
      if ( (ret=block_match_sub(name,&width,&look)) == 0){
            *block_No=look;return(0);}
      if(look==1) if (ret==-1)
         switch(block_match_sub(name,&width,&width)){
            case  0:*block_No=0;return(0);
            case -1:*block_No=0;return(-1);
      }
      *block_No=look;return(ret);
   }
}
int figure_set(char **string){
   char *str;
   int a=0,b=1;
   str=*string;br_skip(str);
   switch (*str){
      case'-':b=-1;
      case'+':++str;
   }
   while ( (*str>='0')&&(*str<='9') ){
      a=a*10+*str-'0';++str;}
   a*=b;*string=str;return(a);
}
void ks_set(char key,char half,char type){
   char i,a,b;
   b=a=key_s[(key+2)%7]=m_key_s[key]+half;
   for (i=key+1;i<=6;i++)
      a=key_s[(i+2)%7]=a+ks_type[type][(i-key-1+7)%7];
   a=b;
   for (i=key-1;i>=0;i--)
      a=key_s[(i+2)%7]=a-ks_type[type][(i-key+7)%7];
}
```

リスト11 ZPP.H

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  マクロセットファイル  */
#include <stdio.h>
#include <string.h>
#define FILEMAX 2000 /*最大読み込み行数*/
#define BLOCKMAX 256 /*ブロック数、必ず2のn乗であること*/
#define LINEMAX 1024 /*一行の最大文字数*/
#define MAJ 10
/*コマンドネーム宣言*/
#define EXT_ON    "#ext_on",7
#define EXT_OFF    "#ext_off",8
#define B_HEAD    '*' /*ブロックコマンドのヘッダ*/
/*マクロ関数宣言*/
#define br_skip(s)          while((*s!=0)&&((*s==32)||(*s=='¥t')))++s
#define str_skip(s,c)       while((*s!=0)&&(*s!=c))++s
#define str_skip2(s,c1,c2)  while((*s!=0)&&(*s!=c1)&&(*s!=c2))++s
#define B_err(str)          error_puts(b_fpoint,str)
#define s_cpy(o,s,e)        while((++s)!=e)*(++o)=*s
```

リスト12 ZPP_HED.H

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  サブルーチン用初期設定ファイル  */
#include "Zpp.h"
/*呼び出し変数宣言*/
extern struct {
   char *name;
   short s_point;
   short e_point;
   short time;
   short flag;
   /*ビット1が再帰呼び出し防止用、2がブ
   ロックでエラーを出したかどうかのフラグ*/
}block[BLOCKMAX+1],*b_sort[BLOCKMAX+1];
extern char *master_buf[FILEMAX];
extern char *macro_buf[26];
extern char line_buf[LINEMAX];
extern short fsize,fpoint,b_No_max,b_sor_max;
   /*fsize-1 の値のところまでバッファを使用している*/
extern short b_fpoint;
extern short b_No;           /*パス2での作業中の行番号*/
extern short work_track;   /*作業中のトラックナンバー*/
extern short debug1,debug2,debug_time;   /*デバッグコマンド用変数*/
extern short por_mod;   /*ポルタメントモード*/
extern short tie_b,tie_p;
extern char key_s[7];         /*キースケール*/
extern char m_key_s[7];          /*Cメジャーのキースケール*/
extern char ks_type[16][7];   /*スケールタイプ*/
extern char key_str[12][4];         /*キーネーム*/
extern char oct,key_no,out_oct,tie,tie_key;
extern char ext[4],buffer[LINEMAX];
extern char modflag,b_set_mod;
extern FILE *fpi,*fpo,*zms;
```

リスト13 ZPP_SUB.H

```
/*  Zpp Z-muSiCファイル プリプロセッサ  */
/*  サブルーチン用呼び出し関数宣言ファイル  */
/*呼び出し関数宣言*/
extern void error_puts(short err_line,char *err_mes);
extern void end();
extern void output(char *outstr);
extern void track_out(char *buf);
extern short block_match ( char *name , short *block_No);
extern int figure_set(char **string);
extern void ks_set(char key,char half,char type);
```

リスト14 MAKEFILE (参考)

```
CC      = gcc
CFLAGS  = -c

if "$(O)" == "on"
CBEST   = -O -fomit-frame-pointer -finline-functions -fstrength-
reduce -fforce-mem -fforce-addr -fcombine-regs -fcaller-saves
endif

AS      = as
AFLAGS  =
LK      = lk
LFLAGS  = -l -x
LLIB    = clib.a floatfnc.a GNUlib.a

zpp.x      : zpp*.o
        $(LK) $(LFLAGS) $^ $(LLIB)

zpp.o      :: zpp.c
        $(CC) $(CFLAGS) $(CBEST) $<

zpp%.o     :: zpp%.c
        $(CC) $(CFLAGS) $(CBEST) $<
```

▶今回，めでたくメモリを増設しました。これで世の中についていけます。

大野 秀樹(20)鹿児島県

おまけ

Z-MUSICでバビンチョ（ゲーム応用編）

Nishikawa Zenji 西川 善司

単に音楽を作るだけでなく，それをアプリケーションのなかで生かしていくことも重要です。ここではZ-MUSICのゲーム関係の機能と使い方を紹介します。来月の特集の前哨戦として参考にしてください。

SION IIを見てもらえればおわかりのようにZ-MUSIC ver.1.10ではゲーム用の音楽ドライバとして使うための機能が大幅に強化されました。ここではZ-MUSICをアセンブラレベルのゲームプログラムから呼び出してBGM演奏させたり，効果音を演奏させたりするためのノウハウを具体的に説明します。この記事を読むためにはZ-MUSICの演奏データに関する知識と68000CPUのアセンブラの知識がある程度必要です。

オプションスイッチいろいろ

ドライバの組み込みは実は基本的ながらも，もっとも重要です。各状況において設定すべきオプションスイッチを，1つひとつピックアップして解説していきましょう。

●組み込みメッセージの制御

－G

なにもせずにZ-MUSICを組み込むと，確保したバッファの容量などの情報が画面に出力されます。また，あのかっこいい（？）Z-MUSICのロゴも表示されます。このスイッチを設定するとこれらを一切表示しないようにすることができます。機能自体に特別な深い意味はないのですがゲームを起動する際の画面バランスに気を遣う人（？）は設定するといいでしょう。

●多重割り込みに対応させる

－M

ゲームのメインプログラムでラスタースクロールやその他のMFPを使った割り込みを設定している場合には設定しましょう。設定しないとZ-MUSICの割り込み処理中は一切の割り込みがキャンセルされてしまいます。しかしこのモードで，MFPを使った割り込み処理が極端に重くなった場合は音楽側のテンポに影響が出ることがあります。ご了承ください。

●初期化制御

－N

Z-MUSICでは新たに演奏データを演奏しようとするとMIDI楽器に初期化パラメータを設定したりFM音源の初期化などを行います。この初期化処理は新たに演奏する曲の都合を無視したものです。たとえばMIDI音源の10チャンネル以下しか使用していない曲を演奏しようというときでも11チャンネル以上の初期化も行ってしまいます。つまり「余計なお世話」をしてくれているわけですね。

この「余計なお世話」はゲームの曲の切り替え時においてゲーム画面のスクロールの一時停止といった目に見える症状になってしまう場合があります。そこでこの‘－N’オプションスイッチを設定すると以後この「余計なお世話」処理を省くことができます。ただしFM音源/MIDI楽器に対して一切の初期化処理を行わなくなるので各演奏データには必要な初期化データを盛り込む必要が出てきます（後述）。

●2通りの演奏データ形式

－U

ゲームの演奏データを設計する場合に2通りのバリエーションが存在します。

ひとつはMIDI対応曲と内蔵音源対応曲を分けてそれぞれ用意する場合です（本誌6月号付録のSION IIやコナミの「出たな!!ツインビー」など）。

もうひとつは，内蔵音源がメインでもMIDIボードがある場合はMIDI音源もこれに重なって鳴るといった，ひとつのデータで両方をまかなうタイプのものです（コナミの「パロディウスだ!」など）。

問題は後者のタイプです。Z-MUSICでは常駐処理のときにMIDIボードの有無を検査してMIDIボードがなければ以後MIDIを使用した曲は頭からはじいて演奏できなくなってしまいます。

これを解決するのが‘－U’オプションです。MIDIボードがない場合はMIDI部の演奏を自動的にミュートして内蔵音源部のみ演奏できるモードになります。もちろん，MIDIボードがある場合にこのスイッチを設定した場合はMIDI部をミュートせず通常に演奏をしてくれます。

●汎用ワークエリアの設定

－Wn

ゲームなどの曲で演奏のたびにAD PCMコンフィギュレーションファイルを読んだりAD PCMの加工処理を行ったりするのは非常識ですから汎用ワークはまったく不要ということになります。ですから普通は，

　　－W0

でいいことになります。

●AD PCMデータ

－B ZPDファイルネーム

－Pn

ゲーム中に用いるすべてのBGMがひとつのZPDデータを用いている場合は，

　　－B ZPDファイルネーム

でドライバ組み込み時に組み込んでしまうとよいでしょう。‘－B’は組み込むZPDファイルのファイルサイズ分自動的にAD PCMバッファを確保してくれますので便利です。

ゲーム中に数回ZPDデータを切り替えたりする場合は‘－P’で必要分バッファを確保してあとからファンクションコールなどを使ってZPDデータやAD PCMデータを登録するということになります。この方法は1曲に必要な最大AD PCMバッファを確保しておけばいいので消費メモリを節約できますが管理が難しくなるのと，曲の切り替わりが必然的に遅くなるのでアクションゲーム系ではあまりオススメできません。しかし，ロールプレイングゲーム（以下RPG）やアドベンチャーゲーム（以下ADV）など，それほど高速処理が要求されないものであればこちらの方法が有用かもしれません。

アクションゲーム系は，やはり多くの市販ソフトも実践しているような，AD PCM

音色やAD PCM効果音をすべてひとまとめにして音源ドライバ側に登録してしまうのがいいでしょう。ちなみにSION IIもそうしていました。

●トラックバッファ

－Tn

曲データをゲーム開始前に一括してメインプログラムの管理領域に読み込んでしまうのが賢明です。そして曲の切り替え時にはあらかじめ読み込んである演奏データの先頭アドレスをZ-MUSICに与えてやればいいのです(具体例は後述)。ということはつまりZ-MUSICのトラックバッファはまったく不要ということですから，

－T0

ということになります。

また，ゲームのプログラムが大きくて一括して読み込めない場合は1ステージ内に必要分を先読みするようにすればいいでしょう（例　メインテーマ，ボステーマ)。

ゲームに負担をかけない演奏データとは?

まず，演奏データ制作者はゲームのプログラムに負担をかけないように心掛けて演奏データを作成しなければなりません。また，曲の切り替えに時間がかからないようにしなければなりません。たとえばスクロールシューティングゲームでステージBGMからボスのテーマに切り替わるときに，スクロールが止まってしまってはちょっとみっともありませんよね。

というわけで曲データはすべて応答の高速であるZMDレベルで管理すべきです。でないと曲を切り替えるたびにコンパイルしなくてはならなくなり，曲の切り替えがとても遅くなりますからね。

ここでは曲の切り替えの早い演奏データを作るためのテクニックについて述べることにします。

●共通コマンドは使わない

FM音源音色の定義コマンドやMIDIエクスクルーシブコマンドなどのトラックに依存しない，いわゆる「共通コマンド」は割り込みで処理されないために曲の切り替え時のウェイトになりかねません。つまりZMSファイルをコンパイルしてZMDを得たときに共通コマンド部分のZMDコードが皆無ならばよいのです。

そこでMMLで簡単に置き換えられる共通コマンドはすべてMMLに置き換えましょう。たとえばテンポコマンド(On)は用いずMMLのテンポコマンドを用いるようにしたり，'.ROLAND_EXCLUSIVE'命令や'(Xn$_1$, n$_2$, n$_3$, …, n$_l$)'などはXコマンドや，@Xコマンドを用いたり……。

また，どうしても置き換わらない，

音色登録

MIDI楽器個別コマンド

などは後述する方法で解決します。

結局ゲームに用いる演奏データには，

初期化コマンド…(I)

ベースチャンネル設定…(Bn)

トラック確保…(M tr，size)

チャンネルアサイン…(A ch，tr)

以外の共通コマンドを用いないようにします。以上の4つのコマンドはコンパイル時に消化され共通コマンドコードとしてのZMDデータに生成されません。

●音色登録は?

演奏データにはさまざまな音色が使用されますが，この音色データを1曲ごとに持たせて演奏の開始時にいちいちシコシコと登録していたのでは曲の切り替わりが遅くなってしまいます（アクションゲームのような極端な高速切り替えが要求されていない，RPGやADVなら別にかまわないでしょうが)。

Z-MUSICではFM音源200音色分(No.1～No.200)の専用バッファを持っており，新たな音色登録が行われたり，ドライバが解除されたりしなければ，この領域はいつまでも保存されています。

ですからそのゲーム中に使用されるFM音色をあらかじめドライバの組み込み時にオプションスイッチ'－S'で転送してしまい演奏データには一切のFM音色データを持たないようにすればいいのです。効果音で用いられるFM音色も一緒に送ってしまいましょう。

SION IIでは音色番号29番以下を効果音の音色，30番以降を曲データの音色データに割り振っており，これらの音色データはZ-MUSIC起動時に転送しておき曲データは一切の音色データは持たずに登録済みの音色を使用していました。

MIDI楽器の音色の場合もFM音源音色の場合と同様です。そのゲーム中で使用する音色データをMIDI楽器へあらかじめ登録してしまいましょう。

音色のセットアップファイルの例をリスト1に示します。このようなものを，

ZMUSIC　－Ssetup.zms

のようにしてドライバ常駐とともに組み込んでしまえばいいのです。

もちろんコンパイルして，

ZMUSIC　－Ssetup.zmd

のようにしてもかまいません。

複数のMIDI楽器に対応している場合はMUSICコンフィギュレーションのような場所で楽器を選ばせ，その時点であらかじめ用意しておいたその楽器の音色データをセレクトして転送するようにすればよいでしょう。

●MIDI楽器個別コマンド

音色データやグローバルなMIDI楽器の設定はゲームが開始される前にあらかじめ登録しておくことで解決しますが，1曲ごとに変えたい設定もあります。たとえばパーシャルリザーブなどです。

楽器個別命令も共通コマンドであるためなんとかMMLへコンバートしたいところです。こういったものはエクスクルーシブメッセージで処理を実現していますから，MMLの@XやXへ変換できなくはありませんが，いちいち楽器のマニュアルを見ながらMMLを書いていくのは少し大変です。

そこでリスト2のようなショートプログラムを提供します。アセンブル，リンク，リロケータブル・コンバートしたものをZMD18.Rとします。

このZMD18.RはZMDデータ中にあるMIDI楽器個別コマンドをMMLに変換するものです。

では，ZMD18.Rの使い方を説明します。

たとえばMUSIC.ZMSというMIDI楽器個別コマンド盛りだくさんのZMSファイルがあったとします。これを，

A>ZMUSIC －C MUSIC

としてMUSIC.ZMDを得ます。

次に，

A>ZMD18 MUSIC

とすると，

```
(T1)@X$F0,$41,$10,$42,......
(T1)@X......
  ⋮
  ⋮
```

といったものが画面に出てくるはずです。これらはMIDI楽器個別コマンドがMMLにコンバートされたものです。

これをリダイレクトしてどこかに保存しMUSIC.ZMSのトラック1へ盛り込んでやります。EOX（$F7）の後ろには若干のウェイトが必要なので休符を入れてやりましょう。

●コメント

演奏データのタイトルや作者名を記述するのに使われている.comment文も演奏時には文字列をスキップする動作を強いられるため，これがウエイトになりかねないので用いないようにします。

●テンポ

曲のテンポは共通コマンド（On）を用いずにMMLのTコマンドを用いて変化させたほうがいいということは，すでに述べたとおりです。

ゲーム中に用いる曲の場合，テンポがステージごとにあまりにも速かったり遅かったりすると効果音もそのテンポにつられて速くなったり遅くなったりします。たとえばある面でBGMがテンポ100で演奏されていてレーザーの発射音が「ちゅーーん・ちゅーーん」だったとします。そして別のステージに突入し，BGMのテンポが倍の200になったとすると「ちゅん・ちゅん」になってしまいます。

これを防止（というか辻褄を合わせる）するのが（Zn）コマンドです。この命令は全音符の絶対音長をデフォルトの192からnへ変更するものです。

テンポが速くなるというのはタイマ割り込みの割り込み間隔が短くなるということですから，この割り込み間隔を一定にしつつ曲のテンポを変えることを考えればいいのです。

先ほどのBGMのテンポが速くなったために効果音も速くなってしまった例は（効果音と，元のBGMが全音符の絶対音長が192基準で作られているならば），テンポ200の曲のほうを全音符の絶対音長を96にしてやれば，テンポ200でも割り込み間隔はテンポ100とまったく同じものとなり，この曲はテンポ200，効果音はテンポ100で演奏され万事解決です。

つまり，タイマ値をだいたい一定（あんまり厳密でなくてかまいませんが）にし，1つひとつの音符の絶対音長の長短でテンポを調整してやるイメージです。テンポMMLコマンドTはこの全音符の絶対音長変更コマンドの（Zn）を考慮してタイマ値を算出してしまうのでタイマ値を直接設定できる@Tコマンドを用いるとわかりやすいでしょうか。

SIONIIでは高橋哲史氏によるMT-32版のBGMはだいたいテンポ120±20程度で作られていましたが，通路面のテーマの「SHOOTING MASTER.ZMS」だけテンポ185でした。そこでこの面だけどうしても効果音も速く聞こえてしまうため，これを解決するためにいままで述べたような処理を施してあります。この曲は全音符の音長に128つまり（Z128）を設定しています。

ところでこの（Zn）コマンドは共通コマンドですからZMDデータとしても生成されてしまいます。共通コマンドの撲滅をいままで謳ってきましたが，このコマンドは

リスト1

```
/                 効果音FM音色データ
(v1,0                                                          /自機ミサイル
/         AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
          60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/         AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
          27, 15,  5,  2,  0,  0,  0,  0,  3,  1,  0
          31, 18, 18,  6,  7,  0,  0,  0,  3,  2,  0
          22, 31,  0, 10,  0, 42,  0,  7,  7,  0,  0
          15, 31,  0,  8,  0,  0,  2,  1,  7,  0,  0)
(v2,0                                                          /敵レーザー
/         AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
          58, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/         AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
          12,  5,  1,  5,  5, 24,  1,  0,  0,  1,  0
          16,  4,  1,  5,  2, 30,  1,  0,  3,  0,  0
          29,  6,  5,  5,  2, 16,  2,  0,  3,  2,  0
           9,  4, 31,  8, 10,  0,  2,  4,  3,  1,  0)

                          :
                          :
                          :
                          :

(v29,0                                                         /敵ダメージ
/         AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
          61, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/         AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
          31,  5,  3,  3,  1, 18,  0,  0,  3,  0,  0
          18,  0,  0,  5,  3,  0,  0,  1,  0,  2,  0
          31,  0,  0,  5,  2,  0,  0,  2,  7,  3,  0
          18,  0,  0,  5,  2,  0,  0,  3,  7,  1,  0)

/                 楽器FM音色データ

/        AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2 AME   BASS 1
(@30,    21,  0,  0,  0,  1, 20,  1,  0,  7,  0,  0
         21,  0,  0,  4,  0,  0,  1,  1,  7,  0,  0
         21,  0,  0,  0,  0, 20,  1,  0,  3,  0,  0
         21,  0,  0,  4,  0,  5,  1,  1,  3,  0,  0
/        AL  FB  OM
          4,  5, 15)

/        AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2 AME   SYNTH1
(@31,    21,  0,  0,  0,  0, 25,  0,  2,  0,  0,  0
         21,  0,  0,  9,  0,  6,  0,  4,  0,  0,  0
         21,  0,  0,  9,  0,  6,  0,  2,  0,  0,  0
         21,  0,  0,  9,  0,  5,  0,  2,  0,  0,  0
/        AL  FB  OM
          5,  7, 15)

                          :
                          :
                          :
                          :

/        AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2 AME   BRASS1
(@100,   19, 17,  0,  8,  1, 24,  1,  1,  1,  0,  0
         19, 13,  0,  8, 15, 37,  1,  2,  0,  0,  0
         19, 17,  0,  8,  1, 36,  1,  1,  0,  0,  0
         19,  4,  5,  8,  2,  0,  1,  1,  2,  0,  0
/        AL  FB  OM
          2,  7, 15)

/------------------------------------------------------------
/                 MIDI楽器音色

/        Chord (Saw&Squ)

.roland_exclusive 16,22 ={8,0,0
          67, 72, 79, 82, 68, 32, 32, 32, 32, 32
           0,  0,  3,  0
/ ** PARTIAL1 **
          36, 53, 11,  1,  2,  0, 36,  9
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
          62, 23, 85
         100, 23,  4, 39,  8, 24
         100,  0,  0,  0, 13, 24, 77, 83,100, 89, 76, 72
          94, 74, 91, 12, 27, 12
           0,  0,  0,  0,  0,  9, 43,100,100,100, 99
/ ** PARTIAL2 **
          36, 48, 11,  1,  3,  0,  0,  0
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
          64, 23, 56
         100, 23,  5,103,  6, 43
         100,  0,  0,  0,  7, 22,100, 98,100, 94, 84, 79
          94, 70, 91, 12, 27, 12
           0,  0,  0,  4, 11, 14, 15,100, 96, 92, 91}

.mt32_patch 1,16 ={2,0,24,52,2,0,1}

/---------------------------
/        Keyboard (Harpsichord)

.roland_exclusive 16,22 ={8,4,0
          72, 65, 82, 80, 83, 73, 67, 79, 82, 68
           2,  5,  3,  0
/ ** PARTIAL1 **
          36, 46, 16,  1,  0, 43,  0,  7
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
           0,  0,  0
           0,  0,  3,  0,  7,  0
           0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0
          87, 77, 91,  6, 27, 12
           3,  0,  0, 23, 60, 83,  0,100, 88, 88, 88
/ ** PARTIAL2 **
          48, 51, 16,  1,  2,  0,100,  7
           0,  0,  0,  0,  0,  0,  0, 50, 50, 50, 50, 50
           0,  0,  0
         100, 16,  9,103,  8,  0
           0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0
         100, 75, 27, 12, 15,  5
           1,  1,  0, 23, 60, 92,  0,100, 88, 88, 88}

.mt32_patch 2,16 ={2,2,24,50,2,0,1}

                          :
                          :
                          :
                          :
```

実行に対して時間を食わないので，まぁ，よしとしましょう。

●初期化

前節で解説したようにZ-MUSICを'－N'オプションをつけて初期化なしモードで常駐している場合，各演奏データに楽器初期化などのコマンドを盛り込まないと前の演奏データの設定状態が受け継がれることになってしまいます。

Z-MUSICでは初期化モードでない通常モードの場合，新たな演奏データの演奏開始直前に以下のようなメッセージを送信しています。

FM音源部
　ノイズモードオフ(Y15，0相当)
AD PCM音源
　特になし
MIDI音源部
　リセットオールコントローラーズ($Bn,$79,$00)
　オムニモードオン($Bn,$7D,$00)
　モノモードオフ($Bn,$7F,$00)
　ローカルオン($Bn,$7A,$7F)
　ファインチューニングニュートラル($Bn,$65,$00,$Bn,$64,$01,$Bn,$06,$40,$Bn,$26,$00)
　コースチューニングニュートラル($Bn,$65,$00,$Bn,$64,$02,$Bn,$06,$40)
　ピッチベンドレンジ＝1オクターブ($Bn,$65,$00,$Bn,$64,$00,$Bn,$06,$0C)
　ボリューム＝63($Bn,$07,$3F)
　パンポット＝中央($Bn,$0A,$40))
　ピッチモジュレーション＝0($Bn,$01,$00)
　ダンパーオフ($Bn,$40,$00)
　ピッチベンドニュートラル($En,$00,$40)

以上から必要なものをピックアップして演奏データに盛り込まなくてはなりません。初期化なしモードとそうでない場合の違いはこれらの初期化動作が行われるか，行われないかの違いになってきます。こうしてみるとFM音源とAD PCM音源は初期化なしモードと通常モードの区別を特に意識する必要はないかもしれません。

演奏データをゲームプログラムなどに取り込む前に一度，初期化なしモードで演奏してみるといいでしょう。それで演奏がおかしかったならどの初期化メッセージが足りないのかを見極めて随時それを盛り込んでいくといいでしょう。

ローランド系のDTM楽器（CMやMT，SCシリーズなど）では楽器のメモリアドレスのある箇所にダミーデータを書き込むと本体内の全パートを初期化してくれる便利な機能があります。これを使えば上のような初期化メッセージをいちいち送らなくても少ない手間で初期化が行えます。いくつか例を挙げておくと，

MT-32/CM-32L/64/32P/500のLA音源部$7F，$00，$00にダミーデータを書き込む実用例
　共通コマンド…．ROLAND_EXCLUSIVE $10,$16= {$7F,$00,$00,$00}
　MML…@I$41,$10,$16 X$7F,$00,$00,$00

SC55/SC155/CM300/CM500のGS音源部$40，$00，$7Fにダミーデータを書き込む実用例
　共通コマンド…．ROLAND_EXCLUSIVE $10,$42= {$40,$00,$7F,$00}
　MML…@I$41,$10,$42 X$40,$00,$7F,$00

（初期化したあとの楽器の状態については各楽器のマニュアルを参照してください）

SION IIではSC-55版の曲には上で示した初期化メッセージのほかにベンドレンジを1オクターブに設定するために@G12を記述しています。SC-55では初期化するとベンドレンジが2半音範囲になってしまうからです。

実際にやってみよう

ここではいままで述べてきた事柄を実際の音楽プログラムに対して行ってみることにしましょう。

リスト4-1のようなミュージックプログラムがあったとしてこれをゲーム向きのデータへ修正していくとします。

まず，コンパイルをしてZMDデータを得ます。

　A>ZMUSIC －C LIST4_1

このZMDデータに対してZMD18.Rを実

リスト4

（4－1）

```
.comment An old picture. by Z.Nishikawa (for SC55/SC155/CM300/CM500)
(i)
(b0)
(m1,2000) (a midi1 , 1)
(m2,2000) (a midi2 , 2)
(m3,2000) (a midi3 , 3)
(m4,2000) (a midi4 , 4)
(m5,2000) (a midi5 , 5)
(m7,2000) (a midi10, 7)
(m8,2000) (a midi10, 8)
(m9,2000) (a midi10, 9)
(m10,2000)(a midi10,10)
(m11,2000)(a midi10,11)
(o125)
.roland_exclusive $10,$42 ={$40,0,$7f,0}   / initialize SC55
.sc55_v_reserve={4,2,6,3, 3,0,0,0 ,0,6,0,0 ,0,0,0,0}/パーシャルリザーブ

(t1)@i$41,$10,$42 @e70,70 @30 v10 @p40        r16[do]
(t2)@i$41,$10,$42 @e80,60 @36 v15             r16[do]
(t3)@i$41,$10,$42 @e70,70 @49 v10 @p99 @k-8   r*7[do]
(t4)@i$41,$10,$42 @e80,80         @p54 @h36   r16[do]@m40
(t5)@i$41,$10,$42 @e80,80         @p74 @h36 r*23[do]@m40
(t7)@i$41,$10,$42 @e80,10 @1           @r1   r16[do]
(t8)                                   @r1   r16[do]
(t9)                                      @r1   r16[do]
(t10)                                  @r1   r16[do]
(t11)                                  @r1   r16[do]

/  DS.GUITAR
(t1) o2 q8 @u99 L1 'e-a-<d-' 'd-g-b' 'g-b<d-' 'e-a-<d-'

/  FRETLESS BASS
(t2) o2 q8 @u110 L1 e- d- >b<  d-

/  STRINGS
(t3) o3 q8 @u99 L1 'e-a-<d-' 'd-g-b' 'g-b<d-' 'e-a-<d-'

/MELODY
(t4)@63 o4 q8 v12 @u110@k0L8r4e-d-c4>b-<d-& d-d-d-cd-e-r4 r4g-g-g-4g-g-
a-g-rd-4&(d-,e-)r4
(t4)i127@48i0 o4 @u120L8r4e-d-c4>b-<d-& d-d-d-cd-e-r4 r4g-fg-4g-g- b-<d
-re-4&(e-,d-)r4
(t4)@4 @m o5 @u110L8r4d-e-g-a-rb-& b-a32b-16.b-4b-a-g-4 a-4g-e-2e-& e-g
-a-g-g32a-16.b-r4L8r4<e-e-d32e-16.&e-4d-& d-&d-2>b-<d-e-& e-&e-2.b-& b-a-g32
a-16.g-a-g-4e-&
(t4)L8e-&e-2>b-<d-e-& e-&e-2>b-<d-d32e-16.& e-&e-4f32g-..e-d->b-& b-a-g
-d32e-..d->b-<e-&  e-1& e-1& e-1& e-1

(t5)@63 o4 q8 v6 @u90  @k12L8r4e-d-c4>b-<d-& d-d-d-cd-e-r4 r4g-g-g-4g-g
- a-g-rd-4&(d-,e-)r4
(t5)i127@48i0 o4 @u120L8r4e-d-c4>b-<d-& d-d-d-cd-e-r4 r4g-fg-4g-g- b-<d
-re-4&(e-,d-)r4
(t5)@4 @m o5 @u99 L8r4d-e-g-a-rb-& b-a32b-16.b-4b-a-g-4 a-4g-e-2e-& e-g
-a-g-g32a-16.b-r4L8r4<e-e-d32e-16.&e-4d-& d-&d-2>b-<d-e-& e-&e-2.b-& b-a-g32
a-16.g-a-g-4e-&
(t5)L8e-&e-2>b-<d-e-& e-&e-2>b-<d-d32e-16.& e-&e-4f32g-..e-d->b-& b-a-g
-d32e-..d->b-<e-&  e-1& e-1& e-1& e-1

/drums
(t7)   o2 q8 v16 @u99  L8c4r4ccr4 rcrcrcc4 c4r4ccrc rcr4rcr4
(t8)   o2 q8     @u110 L8r4d4r4d4 r4d4r4d4 r4d4r4d4 r4d4r4d4
(t9)   o2 q8     @u110 L8<c+*0>@u127|:12 r1 r2^d+d+r:|
(t10)  o2 q8     @u40  L8f+f+f+f+f+f+f+f+ f+f+f+f+f+f+a+a+
(t11)  o3 q8     @u70 L4g+

(t1)[loop]
(t2)[loop]
(t3)[loop]
(t4)[loop]
(t5)[loop]
(t7)[loop]
(t8)[loop]
(t9)[loop]
(t10)[loop]
(t11)[loop]
(p)
```

行しリダイレクトして共通コマンド部分をMMLに変換します。

A>ZMD18 LIST4_1 > LIST4_2

リスト4-2のようなものが得られます。

再びリスト4-1を読み込んで上から順番に共通コマンドをつぶしていきましょう。まずいちばん上の.comment文。これを/して無効化します。ずーっと下に降りていってテンポコマンドがありますがこれを無効化しMMLに直します。

その下のMIDI楽器個別コマンドを/で無効化します。ちょうどこれらの共通コマンドの働きと同じものがZMD18.Rによって生成されたリスト4-2です。

このリスト4-2をリスト4-1のMML部分の先頭に挿入します。前述のようにEOX($F7)のあとには多少のウェイトが必要ですから，休符'r16'を2つあるEOXの後ろに挿入します。このままでは1トラックだけr16×2＝r8分演奏が遅れてしまいますからほかのトラックに辻妻合わせを施します。

こうしてできたのがリスト4-3でこれをコンパイルすれば共通コマンドなしのZMDデータができあがります。

演奏データの演奏

できた曲データ（ZMD）をゲームのメインプログラムから演奏するためには，Z-MUSICを呼び出すための簡単なマネージメントプログラム(つまりは演奏開始ルーチン)が必要です。いろいろな演奏開始ルーチンが考えられますがサンプルとしてリスト5を示します。

この例では各演奏データがmusic0, music1, music2……と各曲がそれぞれ偶数番地に読み込まれているとします。そこでmusic0を演奏したいならばD0.Lに0, music1を演奏したいならばD0.Lに1を入れこのルーチンplay_musicをサブルーチンコールするようにします。これだけで演奏が開始されます。

演奏データに埋め込まれているドライバのバージョンIDをa1レジスタが指標していなければならないというファンクション$11の仕様から演奏データの格納アドレスa1に7を足しています。詳しくはファンクション$11の解説を参照してください。

効果音データの作成

ゲームのBGMに重ねて効果音を鳴らす場合はその演奏データのトラックとチャンネルのアサイン関係をしっかり把握していなければなりません。演奏データのプログラマと効果音データのプログラマが異なる場合には事前に打ち合わせをしておいたほうがいいでしょう。

●効果音データの制約

効果音データも基本的にはBGMデータと同じようにZMSファイルをコンパイルしたZMDデータで表現されます。しかし，BGMデータとの大きな違いは，

1)　共通コマンドを持たない（持てない）
2)　効果音の演奏ワークには初期値がない
3)　以下のMMLコマンドを使用したときの動作が保証されない

・[]系コマンド([DO]～[LOOP]は除く)
・和音コマンド

効果音の演奏制御は，高速応答を要求されるのでパラメータの有効範囲チェックなどは一切行いません。また，2)にもあるとおりワークの初期化も行いませんのでディチューンやボリュームなどの基本パラメータの初期値も不定です。必ず設定するようにしてください。

また，効果音にMMLのテンポコマンドを用いることは禁止していませんが，その効果音が鳴るたびにBGMのテンポが変になるので使用しないほうがいいでしょう。

効果音データの実例をリスト6-1に示します。音色データはすでに述べたように事前に登録してあるものとします。これをコンパイルして得たZMDデータをダンプデータに変換したものがリスト6-2です。ZMDデータにはヘッダ'ZmuSiC'などがつ

リスト4

(4－2)

```
(T1)@X$F0,$41,$10,$42,$12,$40,$00,$7F,$00,$41,$F7
(T1)@X$F0,$41,$10,$42,$12,$40,$01,$10,$06,$04,$02,$06,$03,$03,$00,$00
(T1)@X$00,$00,$00,$00,$00,$00,$00,$00,$17,$F7
```

(4－3)

```
/.comment An old picture. by Z.Nishikawa (for SC55/SC155/CM300/CM500)
(i)
(b0)
(m1,2000) (a midi1 , 1)
(m2,2000) (a midi2 , 2)
(m3,2000) (a midi3 , 3)
(m4,2000) (a midi4 , 4)
(m5,2000) (a midi5 , 5)
(m7,2000) (a midi10, 7)
(m8,2000) (a midi10, 8)
(m9,2000) (a midi10, 9)
(m10,2000)(a midi10,10)
(m11,2000)(a midi10,11)

/(o125)
/.roland_exclusive $10,$42 ={$40,0,$7f,0}  / initialize SC55
/.sc55_v_reserve={4,2,6,3, 3,0,0,0 ,0,6,0,0 ,0,0,0,0} /パーシャルリザーブ

(t1)t125
(T1)@X$F0,$41,$10,$42,$12,$40,$00,$7F,$00,$41,$F7 r16
(T1)@X$F0,$41,$10,$42,$12,$40,$01,$10,$06,$04,$02,$06,$03,$03,$00,$00
(T1)@X$00,$00,$00,$00,$00,$00,$00,$00,$17,$F7 r16
(t1)@g12@i$41,$10,$42 @e70,70 @30 v10 @p40          [do]
(t2)@g12@i$41,$10,$42 @e80,60 @36 v15             r8[do]
(t3)@g12@i$41,$10,$42 @e70,70 @49 v10 @p99 @k-8r*19[do]
(t4)@g12@i$41,$10,$42 @e80,80         @p54 @h36  r8[do]@m40
(t5)@g12@i$41,$10,$42 @e80,80         @p74 @h36r*35[do]@m40
(t7)@g12@i$41,$10,$42 @e80,10 @1           @r1   r8[do]
(t8)                                       @r1   r8[do]
(t9)                                          @r1   r8[do]
(t10)                                      @r1   r8[do]
(t11)                                      @r1   r8[do]
               :
               :
               :
```

リスト5

```
Z_MUSIC macro   func            *ドライバへのファンクションコール
        moveq.l func,d1
        trap    #3
        endm

play_music:
        * < d0.l=曲番号(0~16383)
        movem.l d0-d2/a0-a1,-(sp)
        add.l   d0,d0
        add.l   d0,d0
        movea.l music_data_tbl(pc,d0.l),a1
        addq.w  #7,a1
        moveq.l #0,d2                           *高速応答モード
        Z_MUSIC #$11
        movem.l (sp)+,d0-d2/a0-a1
        rts

music_data_tbl:
        dc.l    music0
        dc.l    music1
        dc.l    music2
        dc.l    music3
        dc.l    music4
        dc.l    music5
        dc.l    music6
        dc.l    music7
                :
                :
                :
                :
```

きますがこれはまったく不要ですので取ってしまいます(残しておいても構いませんがメモリのムダですから)。

そしてリスト6-2の10バイト以降をアセンブラのプログラム形式に変換してやるとリスト6-3のようなものになります。こういったデータをあとはどんどん蓄積してやります。

リスト6-4のように複数トラックの効果音データも作成可能です。これをコンパイルし同様に変換してやるとリスト6-5のようになります。

●効果音の演奏における2つの形式

効果音を鳴らす場合には2通りの形式が存在します。

ひとつは効果音が鳴るチャンネルをあらかじめ空けておく形式です。つまり効果音が鳴るチャンネルをあらかじめ決めておいてBGMデータを作成するわけです。たとえばFM音源の1チャンネルから7チャンネルまでをBGM専用チャンネルとし，8チャンネル目を効果音チャンネルに割り当てる，といった場合がそうです。

2つ目はBGMを演奏しているチャンネルに割り込んで効果音を演奏する形式です。たとえばFM音源の1から8チャンネルすべてのチャンネルでBGMを演奏しそのうち8チャンネルが効果音に切り替わるといった場合がそうです。この場合，効果音が頻繁に演奏されると8チャンネル目はほとんど効果音しか鳴らなくなりBGMの1声が欠けて聞こえることになります。ですからBGMの設計をする際に，効果音が被って

リスト6

(6－1)

```
(v12,0                        /音色データは事前に登録してある
/       AF  OM  WF  SY  SP PMD AMD PMS AMS PAN
        59, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
/       AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AME
        23,  8,  1,  8,  7, 31,  2,  1,  5,  3,  0
        30,  8,  8,  7,  5, 20,  1,  2,  1,  2,  0
        22,  3,  7,  8,  1, 20,  1,  1,  3,  0,  0
        27,  0,  0,  9,  0,  0,  1,  2,  5,  1,  0)


(i)
(m8,1000)
(a8,8)
(t8)@12 o3 q8 v14 p3 @k0 @m60 @h24 @s3 (a16<a)&a4.&(a16>a)
(p)
```

(6－2)

```
                  この部分は不要なので切り捨て
          ┌────────────────────┐
00000000  10 5A 6D 75 53 69 43 11 FF FF 00 01 00 00 00 02 .ZmuSiC........
00000010  00 07 A0 0C B6 05 B3 D1 00 00 00 00 E6 00 3C E8 .. .ｶ.ｳﾑ....  <
00000020  00 18 FF FF D6 00 03 00 00 E0 39 00 0C FF FF 00  ...ﾖ.... 9.....
00000030  00 00 40 00 01 45 48 FF E0 45 00 0C 00 0C 00 00 ..@..EH.潰......
00000040  FF C0 00 FF FF                                   .ﾀ...
```

(6－3)

```
.dc.b   $00,$01                    演奏総トラック数(.W)
.cd.b   $00,$00,$00,$02            演奏データまでのオフセット(.L)
.cd.b   $00,$07                    演奏絶対チャンネル(.W)
.cd.b   $a0,$0c,$b6,$05,$b3,$d1,$00,$00        演奏データ
.dc.b   $00,$00,$e6,$00,$3c,$e8,$00,$18,$ff,$ff,$d6,$00,$03,$00,$00,$e0
.dc.b   $39,$00,$0c,$ff,$ff,$00,$00,$00,$40,$00,$01,$45,$48,$ff,$e0,$45
.dc.b   $00,$0c,$00,$0c,$00,$00,$ff,$c0,$00,$ff,$ff
```

(6－4)

```
(i)
(m7,1000)
(m8,1000)
(a7,7)
(a8,8)
(t7)@7 o7 q8 v15 p3 @k0 @l1e<g>>cd<<e>(f8<f)p0
(t8)@4 o7 q8 v12 p3 @k0 L32(c>c)&(c<e)r1
(p)
```

(6－5)

```
        .dc.b   $00,$02                    演奏総トラック数(.w)
        .cd.b   $00,$00,$00,$08            演奏データまでのオフセット(.L)
        .dc.b   $00,$06                    演奏絶対チャンネル(.W)
        .cd.b   $00,$00,$00,$29            演奏データまでのオフセット(.L)
        .cd.b   $00,$07                    演奏絶対チャンネル(.W)
        .cd.b   $a0,$07,$b6,$02,$b3,$d1,$00,$00        6チャンネル用演奏データ
        .dc.b   $00,$00,$64,$01,$ff,$73,$01,$ff,$54,$01,$ff,$56,$01,$ff,$70,
$01
        .dc.b   $ff,$e0,$65,$00,$18,$00,$18,$00,$00,$00,$20,$00,$01,$b0,$ff
        .cd.b   $a0                                    7チャンネル用演奏データ
        .dc.b   $04,$b6,$0a,$b3,$d1,$00,$00,$00,$00,$e0,$60,$00,$06,$ff,$ff,
$00
        .dc.b   $00,$ff,$80,$00,$ff,$e0,$54,$00,$06,$00,$06,$00,$00,$00,$aa,
$ab
        .dc.b   $01,$80,$c0,$c0,$ff
```

リスト7

```
Z_MUSIC macro   func            *ドライバへのファンクションコール
        moveq.l func,d1
        trap    #3
        endm

se_play1:
        * < d0.l=sound number
        * - all
        movem.l d0-d2/a0-a1,-(sp)
        add.w   d0,d0
        move.w  se_tbl(pc,d0.w),d0
        lea     se_tbl(pc,d0.w),a1
        moveq.l #7,d2                   *割り込むトラック番号
        Z_MUSIC #$12
        movem.l (sp)+,d0-d2/a0-a1
        rts

se_play2:
        * < d0.l=sound number
        * < d2.l=ch number
        * - all
        movem.l d0-d2/a0-a1,-(sp)
        add.w   d0,d0
        move.w  se_tbl(pc,d0.w),d0
        lea     se_tbl(pc,d0.w),a1
        move.b  d2,7(a1)                *絶対チャンネルセット
        addq.b  #1,d2                   *割り込むトラック番号
        Z_MUSIC #$12
        movem.l (sp)+,d0-d2/a0-a1
        rts

se_play3:
        * < d0.l=sound number
        * - all
        movem.l d0-d2/a0-a1,-(sp)
        add.w   d0,d0
        move.w  se_tbl(pc,d0.w),d0
        lea     se_tbl(pc,d0.w),a1
        lea     which_ch(pc),a0
        eori.b  #1,(a0)
        move.b  (a0),d2
        addq.b  #6,d2
        move.b  d2,7(a1)                *絶対チャンネルセット
        addq.b  #1,d2                   *割り込むトラック番号
        Z_MUSIC #$12
        movem.l (sp)+,d0-d2/a0-a1
        rts

which_ch:       dc.w    0

se_tbl:
        dc.w    se0-se_tbl              *$00
        dc.w    se1-se_tbl              *$01
        dc.w    se2-se_tbl              *$02
        dc.w    se3-se_tbl              *$03
        dc.w    se4-se_tbl              *$04
                    :
                    :
                    :

se0:
        .dc.b   $00,$01                 演奏総トラック数(.W)
        .dc.b   $00,$00,$00,$02         演奏データまでのオフセット(.L)
        .dc.b   $00,$07                 演奏絶対チャンネル(.W)
        .dc.b   $a0,$0c,$b6,$05,$b3,$d1,$00,$00     演奏データ
        .dc.b   $00,$00,$e6,$00,$3c,$e8,$00,$18,$ff,$ff,$d6,$00,$03,$00,$00,$e0
        .dc.b   $39,$00,$0c,$ff,$ff,$00,$00,$00,$40,$00,$01,$45,$48,$ff,$e0,$45
        .dc.b   $00,$0c,$00,$0c,$00,$00,$ff,$c0,$00,$ff,$ff
        .even
se1:
        .dc.b   $00,$02                 演奏総トラック数(.w)
        .dc.b   $00,$00,$00,$08         演奏データまでのオフセット(.L)
        .dc.b   $00,$06                 演奏絶対チャンネル(.W)
        .dc.b   $00,$00,$00,$29         演奏データまでのオフセット(.L)
        .dc.b   $00,$07                 演奏絶対チャンネル(.W)
        .dc.b   $a0,$07,$b6,$02,$b3,$d1,$00,$00     6チャンネル用演奏データ
        .dc.b   $00,$00,$64,$01,$ff,$73,$01,$ff,$54,$01,$ff,$56,$01,$ff,$70,$01
        .dc.b   $ff,$e0,$65,$00,$18,$00,$18,$00,$00,$00,$20,$00,$01,$b0,$ff
        .dc.b   $a0                                 7チャンネル用演奏データ
        .dc.b   $04,$b6,$0a,$b3,$d1,$00,$00,$00,$00,$e0,$60,$00,$06,$ff,$ff,$00
        .dc.b   $00,$ff,$80,$00,$ff,$e0,$54,$00,$06,$00,$06,$00,$00,$00,$aa,$ab
        .dc.b   $01,$80,$c0,$c0,$ff
        .even
                    :
                    :
                    :
                    :
```

くるチャンネルにはメロディやベースなどの重要なパートを割り当てないように気をつけるべきでしょう。

●1番目の形式のお話

リスト7を見てください。効果音のマネージメントプログラムの一例です（リスト7中のサンプル効果音は，それぞれリスト6-3，リスト6-5とまったく同じものです)。

まずse_play1を見てください。

これはトラック1から6までをFM音源1チャンネルから6チャンネルに割り当ててこれをBGMで使用し，FM音源7～8チャンネルを効果音として使用する場合を考えます。たとえばd0.lに効果音番号0を入れてse_play1を呼ぶと，トラック7番からFM音源第7チャンネルでse0のデータが演奏されます。se_play1中の，

```
moveq.l #7,d2
```

というのは，何番トラックで効果音を演奏するかを設定しています。いま，トラック1～6まででBGMを演奏しているわけですからトラック7は未使用です。よってなんにもジャマされず，ごく普通に演奏がなされます。

なぜこの例で効果音se0がFM音源第7チャンネルで演奏されるかというと，それはse0の効果音データに演奏絶対チャンネル番号が6 (0～7のうちの6)というのが埋め込まれているからです(囲み参照)。

同様の条件で，se1を鳴らす場合も同様です。se1は複数トラックによって構成された効果音データです。

```
moveq.l #7,d2
```

とあるからトラック7から演奏するわけです。まず，se1の第1トラックはd2.lで指定したとおりトラック7で演奏されます。次にse1の第2トラックはd2.lで指定したトラック番号＋1のトラック8で演奏されます。

このように複数トラックで構成された効果音はd2.lで指定したトラック番号から順番にトラックが割り当てられ演奏されます。

●2番目の形式のお話

では今度はトラック1～8がFM音源1～8チャンネルにアサインされBGMが演奏されていたとして，これに効果音を鳴らす場合を考えます。こちらは前節でいった2番目の形式です。

se_play1は結論からいえば，このままこの形式に対応できます。単一トラックのse0も複数トラック構成のse1も大丈夫です。ちゃんと効果音が鳴っているあいだはそのチャンネルはBGM演奏を中断し，効果音を演奏して，演奏し終わるとなにごともなかったようにそのチャンネルのBGM演奏を再開するはずです。

では，

```
moveq.l #7,d2
```

を，

```
moveq.l #1,d2
```

にしてse0を鳴らしたとしたらどうなるでしょうか。つまりse0がBGMのトラック1から割り込むという場合です。

確かにBGMのトラック1（FM音源第1チャンネル）の演奏が中断されますが，効果音se0はFM音源第7チャンネルで演奏されることになっていますので，そのチャンネルで効果音を演奏しようとすると，BGM側のFM音源第7チャンネルにアサインされたトラック7がFM音源第7チャンネルをアクセスしてきます。いい換えると複数の違った演奏データを同時にたった1個のFM音源チャンネルで演奏しようとする状況が巻き起こるわけです。

つまりこの形式ではひとつの大前提があるのです。これから演奏する効果音のチャンネルが，割り込むトラックにアサインされているチャンネルと同一でなければならないのです。

この部分はたいへん重要です。

se1は2本のトラックで構成された効果音ですがこの2本目のトラックがFM1チャンネルに割当てられていたとします。つまりse1の上から5段目が$00,$07でなくて$00,$00となっている場合です。これを，

```
moveq.l #7,d2
```

で7トラックから割り込ませるとします。

さて実行されるとBGMのトラック7に，FM音源第7チャンネルにアサインされたse1の第1トラックが割り込んできます。以後FM第7チャンネルは効果音を演奏します。FM音源第7チャンネルにアサインされたBGM側のトラックは効果音に占領されているトラック7しかないので，誰も効果音演奏をジャマせずうまくいきます。

そして次にBGM側のトラック8に，FM音源第1チャンネルにアサインされたse1の第2トラックが割り込んできます。BGM側のトラック8はそれまでFM音源第8チャンネルで演奏していたBGMを中断しFM音源第1チャンネルで効果音を演奏し始めます。

しかし，その後，BGM側のトラック1がFM音源第1チャンネルをアクセスしてくるのでおかしなことになってきます。

つまりse1の第1トラックは正常に演奏されるが，第2トラックは正常に演奏されないのです。

この話を図式化したものを図1に示します。

●その他の応用

効果音データはその演奏絶対チャンネルを持っていました。ですから同じ効果音を違ったチャンネルで鳴らすためには，その絶対チャンネルの違ったデータを別に持たなければならないのでしょうか。それでも構いませんがちょっとメモリがもったいないので，効果音データのその絶対チャンネルが書かれている部分を書き換えてひとつの効果音データを流用してやりましょう。

リスト7のse_play2はその一例です。パラメータは効果音番号と演奏チャンネルの2つ。効果音データのアドレスを得たあとそのアドレス＋7に演奏希望絶対チャンネルを書き込んでいます。効果音が複数トラックで構成されている場合は，たとえばse1ならば(効果音データのアドレス＋7)のほかに(効果音データのアドレス＋13)にも絶対チャンネルを設定しなければなりません。

se_play2では割り込むトラックはその絶対チャンネル＋1の値にしています。これはBGMのトラック1～8がFM音源の第1チャンネルから第8チャンネル(絶対チャンネルにして0～7)にアサインされていることを前提としています。

se_play3はse_play2を少し改良したものです。これは効果音をFM音源第7チャンネルと第8チャンネルを交互に用いて鳴らす処理をつけてみたものです。こちらもBGMのトラック1～8がFM音源の第1チャンネルから第8チャンネル(絶対チャンネルにして0～7)にアサインされていることを前提としています。

こういった状況に応じて効果音データを書き換えるような処理を工夫するといろいろと面白いことが実現可能です。SIONⅡでは効果音データのパンポットをMAGICのX，Zパラメータによって随時設定し2Dサウンドシステムを実現しています(たいしたものではないんですけどね)。興味のある人は6月号の付録ディスクのSIONⅡのソースリストの中のse.sをのぞいてみまし

絶対チャンネル

内部表現の絶対チャンネルはベースチャンネルに無関係に操作対象デバイスを決定するものです。Z-MUSICのマニュアルにも書いてありますが，絶対チャンネル0～7がFM音源の1～8，絶対チャンネル8がAD PCM1，絶対チャンネル9～24がMIDI1～16。そしてPCM8独立チャンネルモード時には絶対チャンネル25～31がAD PCM2～8に対応します。

ょう。

●MIDI対応曲にFMの効果音

いままではFM音源のBGMにFM音源の効果音を割り込ませる話でしたが，MIDI対応の曲にFM音源の効果音を割り込ませるにはどうしたらよいのでしょうか。

BGMをMIDIで，効果音はFM音源で，という場合は各音源の目的がすでに区別されているので，効果音の使用チャンネルがBGMの使用チャンネルとかち合わないので形式1のように実現できます。

たとえば，トラック1～16までをMIDIでBGMに使用している場合なら効果音は未使用トラックであるトラック17以降に割り込ませれば(割り込んではいないのですが)よいことになります。この場合はわかりやすく管理も楽ですね。

では，FM音源＋MIDI楽器でBGMを，そしてFM音源で効果音を，という場合はどうでしょうか。なんか複雑そうですがこれは結局FM音源のみのBGMにFM音源の効果音を割り込ませた方法がほとんどそのまま使えます。

たとえば，FM音源の第1チャンネルから第6チャンネルにアサインしたトラック1～6，そしてMIDIにアサインしたトラック7～12でBGMを演奏していたとします。つまりFM音源第7，8チャンネルは未使用というときに，この空いている2つのチャンネルを使って効果音を鳴らす場合を考えます。これはチャンネルがまったくかち合わないことから未使用トラックで効果音を鳴らせばよいので形式1がそのまま適用できます。よって効果音はトラック13以降へ割り込ませてやれば（割り込んではいませんが）よいことになります。

ではFM音源第1チャンネルから第8チャンネルまでをトラック1～8にアサインし，MIDIをトラック9～16にアサインし，これでBGM演奏して，これにFM音源の効果音を割り込ませる場合を考えます。FM音源の効果音はMIDIに対してなんの影響も与えないことからトラック9～16を除外して考えていいわけですから，そうなると話はまったく形式2と同じになってきます。ちょっとくどい感じがしますが一応例を示しておきますと，効果音をFM音源第8チャンネルで演奏するならトラック8に割り込ませればいいことになります。

●MIDIの効果音

MIDIで効果音を鳴らす場合にはもっと話が単純化します。ただMIDIチャンネルというものは単一チャンネルでもFM音源チャンネル複数分として動作しますから，ひとつのMIDIチャンネルに対して複数のトラックをアサインするケースが頻繁に出てきます。というわけで形式1，2ともたいした区別はなくなります。

しかし，管理の容易さから形式1をオススメします。つまり効果音専用のMIDIチャンネルを取っておいてこのチャンネルを用いて効果音を鳴らすようにするわけです。

●AD PCM音の効果音

もう，ここまでの話を理解してしまった人ならばちょっとくどいですかね。

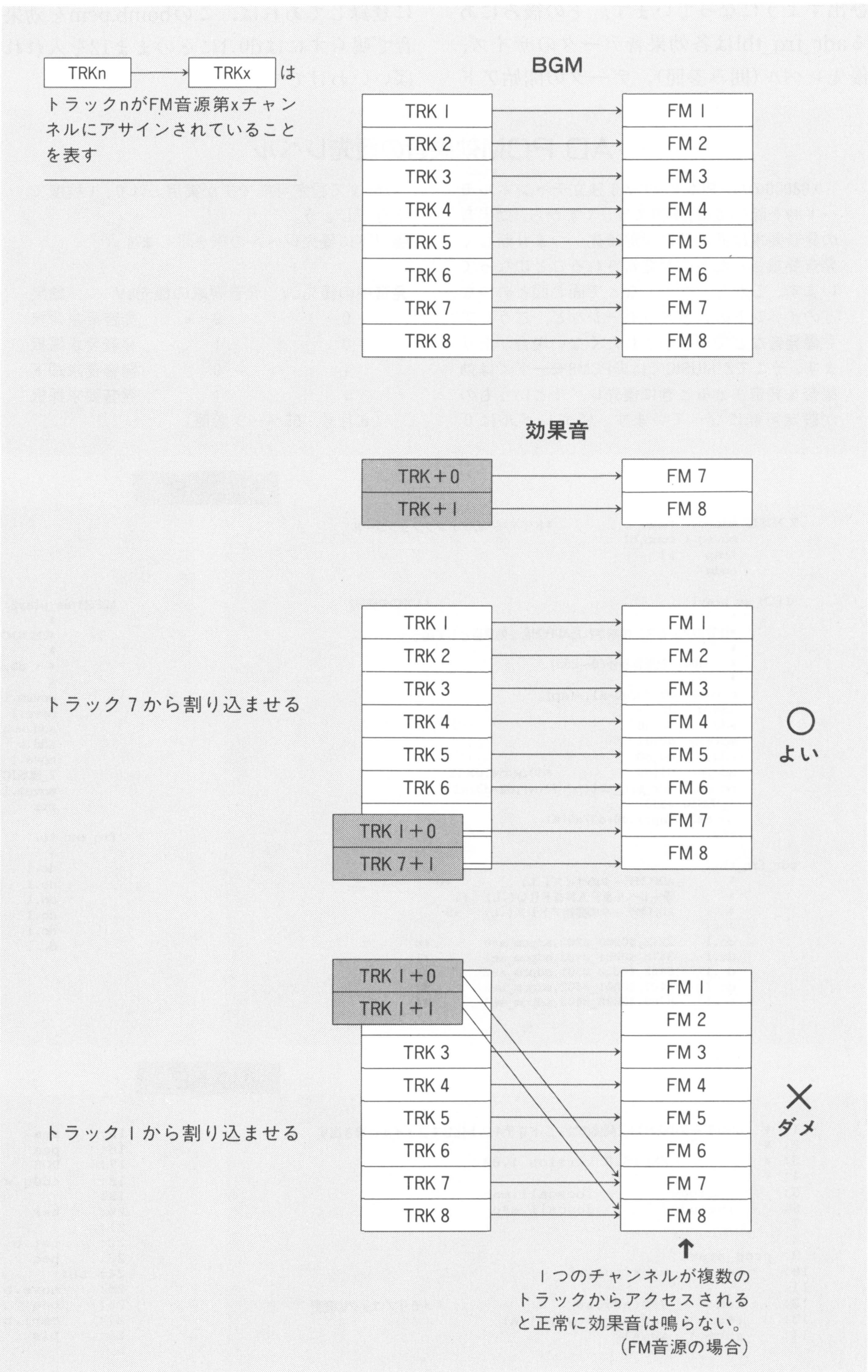
AD PCMはPCM8.Xモードでないときは単音ですから形式2のみ有効ですね。

FM音源とAD PCMが連続するようなチャンネルアサイン構造にしておいて，FM音源とAD PCM音源を両方使った効果音なんていうのも演奏できますね。

また，複雑なものでないワン・ショットのAD PCMの効果音の演奏ならば専用のファンクションが設けてありますので，それを用いたほうが高速です。

リスト8にその一例を示します。

図1　効果音モード



まず，リスト中のADPCM_se_play1を見てください。これは任意のアドレスに格納されているAD PCMデータをファンクションコール$13を使って，BGMに割り込んで鳴らすためのプログラムです。BGM側でAD PCMをドラムなどで使用している場合はそれを一時停止して効果音を鳴らします。効果音が鳴り終わると自動的に演奏に戻ります。

ADPCM_se_play1ではアドレスに配置されたデータに便宜的に番号をつけ，呼び出すときにはd0.lにその番号を代入して呼び出すようになっています。その後ろにあるadr_frq_tblは各効果音データのサイズ，優先レベル（囲み参照），データの開始アドレスの管理テーブルです。

ADPCM_se_play2はZ-MUSICにすでにZPDデータなどによって登録済みのAD PCM音を効果音で鳴らすためのサブルーチンです。ファンクションコール$14をそのまま使っただけの単純なものです。d0.lに指定する効果音番号はAD PCMコンフィギュレーションで設定したノートナンバーに相当します。たとえばコンフィギュレーションで，

12=bomb.pcm

のように書き，これをZPDとしてZ-MUSICに登録してあれば，このbomb.pcmを効果音で鳴らすにはd0.lにそのまま12を入れればいいわけです。

後ろについているfrq_pan_tblというテーブルはADPCM_se_play1のadr_frq_tblからデータサイズとデータアドレスを取り去ったような形状をしています。

＊　　＊　　＊

と，ちょっとわかりにくいところもあったと思いますが，Z-MUSIC ver.1.10の解説はひとまずこれでおしまいです。では皆さん増刷版「Z-MUSICシステム」でお会いいたしましょう。

AD PCM効果音の優先レベル

X68000のAD PCMは(PCM 8 独立チャンネルモード時を除いては)単声です。ですから，効果音の発音要求は基本的に後着優先，つまり新しく発音要求されたものが発音されることになっています。しかし，ゲームなどで面と面とのつなぎのイベントの音声メッセージなど，どうしても爆発音などでかき消されたくない場合があります。そこでZ-MUSICでは非PCM8モードでは効果音を発音させるときに優先レベルというものが設定可能になっています。優先レベルは 0 〜255まで設定可能ですが実用上は 0，1 程度で十分でしょう。

●以下に優先レベルの例を示します。

発音中の優先LV	発音要求の優先LV	結果
0	0	発音要求採択
0	1	発音要求採択
1	0	発音要求却下
1	1	発音要求採択

（8 月号，55ページ参照）

COMCHKコマンド

開発途中はいろいろと曲のデータのほうも日に日に手を入れられ改良されていくでしょうから，共通コマンドを撲滅するのは最終的な段階でかまわないでしょう。SION II もマスターアップ直前まで各曲データに共通コマンドが存在していました。

そこで最終的なチェック用としてリスト 3 にコンパイルして生成したZMDデータに共通コマンドがあるかないかを検査するプログラムを提供します。

リスト 3 をアセンブル，リンク，リロケータブルコンバートするとCOMCHK.Rという実行ファイルができます。使い方は，

COMCHK ZMDファイル

とするだけです。共通コマンドがあれば「ある」と，なければ「ない」という結果を出力してくれます。

リスト8

```
Z_MUSIC macro   func            *ドライバへのファンクションコール
        moveq.l func,d1
        trap    #3
        endm

ADPCM_se_play1:                                   *ADPCM効果音
        *
        *任意のアドレスに格納されたADPCM音を効果音として鳴らす
        *
        * < d0.l=効果音番号(0～255)
        *
        movem.l d0-d3/a0-a1,-(sp)
        add.w   d0,d0
        add.w   d0,d0
        move.w  d0,d1
        add.w   d0,d0
        add.w   d1,d0           *d0.w=d0.w×12
        movem.l adr_frq_tbl(pc,d0.w),d2-d3/a1
        Z_MUSIC #$13
        movem.l (sp)+,d0-d3/a0-a1
        rts

adr_frq_tbl:
        *       ADPCMデータのサイズ(.L)         +0
        *       優先レベル&PAN&FRQ(.L)   +4
        *       ADPCMデータの開始アドレス(.L)    +8
        *
        dc.l    2212,$0000_0203,adpcm_se0       *0
        dc.l    3478,$0001_0403,adpcm_se1       *1
        dc.l    5453,$0000_0301,adpcm_se2       *2
        dc.l    1401,$0001_0402,adpcm_se3       *3
        dc.l    8799,$0000_0403,adpcm_se4       *4
                      :
```

```
ADPCM_se_play2:
        *
        *ZMUSICに登録したADPCM音を効果音として鳴らす
        *
        * < d0.l=効果音番号(0～255)
        *
        movem.l d0-d3/a0-a1,-(sp)
        move.l  d0,d2
        add.w   d0,d0
        add.w   d0,d0
        move.l  frq_pan_tbl(pc,d0.w),d3
        Z_MUSIC #$14
        movem.l (sp)+,d0-d3/a0-a1
        rts

frq_pan_tbl:                             *効果音番号
        *       優先レベル&PAN&FRQ(.L)
        dc.l    $0000_0203               *0
        dc.l    $0000_0303               *1
        dc.l    $0000_0201               *2
        dc.l    $0001_0402               *3
        dc.l    $0001_0203               *4
        dc.l    $0001_0103               *5
                   :
                   :
                   :
                   :
```

リスト2

```
*   ZMDデータ中のMIDI関係のコマンドをテキスト化してファイルに書き出す
*
*           ZMD18.R version 1.00
*
    .include        iocscall.mac
    .include        doscall.mac


prog_start:
    lea     work(pc),a6

    lea     $10(a0),a0                  *メモリブロックの変更
    lea     end_of_prog(pc),a1
    suba.l  a0,a1
```

```
        pea     (a1)
        pea     (a0)
        DOS     _SETBLOCK
        addq.w  #8,sp

        bsr     print_title

        tst.b   (a2)+
        beq     print_help              *簡易ヘルプを表示
LL1:
        move.b  (a2)+,d0
        beq     print_help
        cmpi.b  #' ',d0
        bls     LL1
```

```
	subq.w	#1,a2

	lea	sorce_name(pc),a0
LL2:
	move.b	(a2)+,d0
	cmpi.b	#' ',d0
	bls	LL3
	move.b	d0,(a0)+
	bra	LL2
LL3:
	clr.b	(a0)+			*end
	subq.w	#1,a2

	bra	main_ope

kakuchoshi:				*拡張子を設定
	* < a0=filename address
	* X a0
	moveq.l	#91-1,d0
kkchs_lp:
	move.b	(a0)+,d0
	beq	do_kkchs
	cmpi.b	#'.',d0
	beq	find_period
	dbra	d0,kkchs_lp
do_kkchs:
	subq.l	#1,a0
	move.b	#'.',(a0)+
	move.b	#'Z',(a0)+
	move.b	#'M',(a0)+
	move.b	#'D',(a0)+
	clr.b	(a0)
	rts
find_period:
	cmpi.b	#' ',(a0)
	bls	do_kkchs		*'.'はあっても拡張子がないケース
	rts

main_ope:
	lea	sorce_name(pc),a0
	bsr	kakuchoshi

	lea	sorce_name(pc),a2
	bsr	fopen
	tst.b	d5
	bmi	open_err
	bsr	read			*>d3.l=size a5.l=address
	bsr	process
	DOS	_EXIT

process:
	* < a5.l=sorce zmd data address
	* < a4.l=書き込み先バッファ
	move.l	(a5)+,d0
	cmpi.l	#$105a_6d75,d0	*ヘッダチェック
	bne	unid_err
	move.l	(a5)+,d0
	clr.b	d0
	cmpi.l	#$5369_4300,d0	*ヘッダチェック
	bne	unid_err
	moveq.l	#0,d2		*clear flag
pcd_lp01:
	move.b	(a5)+,d1
	bmi	exit_ccd		*共通コマンドエンド
	cmpi.b	#$42,d1
	beq	cnv_mclk		*全音符カウンタセット
	cmpi.b	#$15,d1
	beq	cnv_scm			*ベースチャンネル切り換え
	cmpi.b	#$04,d1
	beq	cnv_vset		*ボイスセット
	cmpi.b	#$05,d1
	beq	cnv_tempo		*テンポ
	cmpi.b	#$18,d1
	beq	cnv_mdtrns		*MIDI生データ転送
	cmpi.b	#$1b,d1
	beq	cnv_vset2		*ボイスセット2
	cmpi.b	#$40,d1
	beq	skip_until0		*adpcm data read
	cmpi.b	#$60,d1
	beq	skip_until0		**com0:read opmd.cnf
	cmpi.b	#$61,d1
	beq	skip_until0		**com2:print message
	cmpi.b	#$62,d1
	beq	cnv_midi		**com22:trans midi data dump
	cmpi.b	#$63,d1
	beq	cnv_abdt		**com35:read adpcm block data
	cmpi.b	#$7e,d1
	beq	pcd_lp01		*dummy code
	cmpi.b	#$7f,d1
	beq	skip_until0		**com36:comment
	bra	unid_err		*unknown command
exit_ccd:
	rts

skip	macro	n
	if	n<=8	then
	addq.w	#n-1,a5
	else
	lea	n-1(a5),a5
	endif
	bra	pcd_lp01
	endm

cnv_vset:	*音色設定
	skip	57

cnv_tempo:	*テンポ
	skip	3

cnv_mclk:	*全音符カウンタセット
	skip	6

cnv_scm:	*ベースチャンネル切り換え
	skip	2

cnv_mdtrns:			*MIDI生データ転送
	move.b	(a5)+,d3
	lsl.w	#8,d3
	move.b	(a5)+,d3		*get how many
cmlp:
	move.b	(a5)+,d1
	tst.b	d2
	bne	LL4
	moveq.l	#16,d2
	pea	atx(pc)
	DOS	_PRINT			*@X
	addq.w	#4,sp
LL4:
	bsr	wrt_lb
	subq.w	#1,d2
	beq	LL5
	cmpi.b	#$f7,d1
	bne	camma
	moveq.l	#0,d2
LL5:
	pea	crlf(pc)
	DOS	_PRINT
	addq.w	#4,sp
	subq.w	#1,d3
	bne	cmlp
	bra	pcd_lp01
camma:
	move.w	#',',-(sp)
	DOS	_PUTCHAR
	addq.w	#2,sp
	subq.w	#1,d3
	bne	cmlp
	bra	pcd_lp01

wrt_lb:
	move.w	d1,-(sp)
	move.w	#'$',-(sp)
	DOS	_PUTCHAR
	addq.w	#2,sp
	move.b	d1,d0
	lsr.b	#4,d0
	add.b	#'0',d0
	cmpi.b	#'9',d0
	bls	LL6
	addq.b	#7,d0
LL6:
	move.w	d0,-(sp)
	DOS	_PUTCHAR		*upper
	addq.w	#2,sp
	andi.w	#$0f,d1
	add.b	#'0',d1
	cmpi.b	#'9',d1
	bls	LL7
	addq.b	#7,d1
LL7:
	move.w	d1,-(sp)
	DOS	_PUTCHAR		*lower
	addq.w	#2,sp
	move.w	(sp)+,d1
	rts

cnv_vset2:	*ボイスセット2
	skip	50

cnv_midi:	**com22:trans midi data dump
cnv_abdt:	**com35:read adpcm block data
skip_until0:
	moveq.l	#0,d2
	move.b	d1,(a4)+
adbt_lp:
	move.b	(a5)+,(a4)+
	bne	adbt_lp
	bra	pcd_lp01

fopen:					*ファイルのオープン(環境変数参照モード)
	* < a2=file name
	* > d5=file handle (error:d5<0)
	* - all
	movem.l	d0/a0-a2,-(sp)

	clr.w	-(sp)
	pea	(a2)
	DOS	_OPEN
	addq.w	#6,sp
	move.l	d0,d5			*d5.w=file handle
	bpl	exit_fopen		*no problem

	movea.l	a2,a0			*a0=file name only(non pathed)

	pea.l	env_work(pc)
	clr.l	-(sp)
	pea	getname(pc)
	DOS	_GETENV
	lea	12(sp),sp
	tst.l	d0
	bmi	exit_fopen

	lea.l	env_work(pc),a1
fopen_lp01:
	lea	open_fn(pc),a2
fopen_lp02:
	move.b	(a1)+,d0
	beq	do_fopen
```

```
	cmpi.b	#';',d0
	beq	do_fopen
	move.b	d0,(a2)+
	bra	fopen_lp02
do_fopen:
	moveq.l	#'¥',d0
	cmp.b	-1(a2),d0
	beq	sva0
	cmp.b	(a0),d0
	beq	sva0_
	move.b	d0,(a2)+
sva0:
	cmp.b	(a0),d0			*'¥'が続いた場合
	bne	sva0_
	addq.w	#1,a0
	bra	sva0
sva0_:
	pea	(a0)
do_fopenlp:
	move.b	(a0)+,(a2)+
	bne	do_fopenlp
	clr.b	(a2)
	move.l	(sp)+,a0

	clr.w	-(sp)
	pea	open_fn(pc)
	DOS	_OPEN
	addq.w	#6,sp
	move.l	d0,d5			*d5.w=file handle
	bpl	exit_fopen		*no problem
	tst.b	-1(a1)			*まだ環境変数が残ってるか
	bne	fopen_lp01
exit_fopen:
	movem.l	(sp)+,d0/a0-a2
	rts

read:
	* < d5.l=file handle
	* > a5=data address
	* > d3.l=size
	* X d0
	move.w	#2,-(sp)		*ファイルの長さを調べる
	clr.l	-(sp)
	move.w	d5,-(sp)
	DOS	_SEEK
	addq.w	#8,sp			*d0.l=file length
	move.l	d0,d3			*d3=length
	beq	fsize0			*file size=0
	bmi	fsize0			*file size=minus

	move.l	d0,-(sp)
	DOS	_MALLOC
	addq.w	#4,sp
	tst.l	d0
	bmi	out_mem			*OUT OF MEMORY
	move.l	d0,a5

	clr.w	-(sp)			*ファイルポインタを元に戻す
	clr.l	-(sp)
	move.w	d5,-(sp)
	DOS	_SEEK
	addq.w	#8,sp

	move.l	d3,-(sp)		*push size
	pea	(a5)			*push addr
	move.w	d5,-(sp)		*file handle
	DOS	_READ
	lea	10(sp),sp
	tst.l	d0
	bmi	read_err		*読み込み失敗

	move.w	d5,-(sp)		*close
	DOS	_CLOSE
	addq.l	#2,sp
	rts

print_title:
	move.w	#2,-(sp)
	pea	title(pc)
	DOS	_FPUTS
	addq.w	#6,sp
	rts

print_help:
	lea	help_mes(pc),a1
	bsr	mes_prt
bye_bye:
	DOS	_EXIT

err_bye:
	bsr	mes_prt
	move.w	#-1,-(sp)
	DOS	_EXIT2

prt_sfn:
	lea	sorce_name(pc),a1
	bra	mes_prt

read_err:
	bsr	prt_sfn
	lea	read_err_mes(pc),a1
	bra	err_bye

out_mem:
	lea	out_mem_mes(pc),a1
	bra	err_bye

open_err:
	bsr	prt_sfn
	lea	open_err_mes(pc),a1
	bra	err_bye

fsize0:
	lea	fsize0_mes(pc),a1
	bra	err_bye

unid_err:
	bsr	prt_sfn
	lea	unid_err_mes(pc),a1
	bra	err_bye

mes_prt:
	pea	(A1)
	DOS	_PRINT
	addq.w	#4,sp
	rts

work:
title:		dc.b $1b,'[35mZMD18.R',$1b,'[m version 1.00 (C) 1992 '
		dc.b $1b,'[36mZENJI SOFT',$1b,'[m',13,10,0
help_mes:	dc.b $1b,'[37m(使い方)',13,10
		dc.b $1b,'[m ZMD18 <読み込みファイルネーム[.ZMD]>',13,10
		dc.b $1b,'[37m(機能)',13,10
		dc.b $1b,'[m'
		dc.b '  ZMDファイル中のMIDI出力関係の共通コマンド',13,10
		dc.b 'をMMLの@Xコマンドに変換してテキスト出力する。',13,10
		dc.b 'MIDI出力をしていないZMDデータには全く無意味。',13,10
		dc.b 'もちろんZMSファイルに対しても無意味。あしからず',13,10
		dc.b 0
read_err_mes:		dc.b 'が読めなかったよ',13,10,0
out_mem_mes:		dc.b 'メモリが不足しているよ! 増設しよう',13,10,0
open_err_mes:		dc.b 'が見つからなかったよ',13,10,0
fsize0_mes:		dc.b 'ファイルサイズが変だ!!',13,10,0
unid_err_mes:		dc.b 'はZMDデータじゃないみたいだね'
crlf:			dc.b 13,10,0
getname:		dc.b 'zmusic',0
atx:			dc.b '(T1)@X',0
	.bss
sorce_name:ds.b	91
open_fn:	ds.b	91
env_work:	ds.b	256
	.even
end_of_prog:
```

リスト3

```
*	ZMDデータ中に共通コマンドがあるか検査する
*
*			COMCHK.R
*
	.include	iocscall.mac
	.include	doscall.mac


prog_start:
	lea	work(pc),a6

	lea	$10(a0),a0			*メモリブロックの変更
	lea	end_of_prog(pc),a1
	suba.l	a0,a1
	pea	(a1)
	pea	(a0)
	DOS	_SETBLOCK
	addq.w	#8,sp

	bsr	print_title

	tst.b	(a2)+
	beq	print_help			*簡易ヘルプを表示
LL1:
	move.b	(a2)+,d0
	beq	print_help
	cmpi.b	#' ',d0
	bls	LL1
	subq.w	#1,a2

	lea	sorce_name(pc),a0
LL2:
	move.b	(a2)+,d0
	cmpi.b	#' ',d0
	bls	LL3
	move.b	d0,(a0)+
	bra	LL2
LL3:
	clr.b	(a0)+				*end
	subq.w	#1,a2

	bra	main_ope

kakuchoshi:					*拡張子を設定
	* < a0=filename address
	* X a0
	moveq.l	#91-1,d0
kkchs_lp:
	move.b	(a0)+,d0
	beq	do_kkchs
	cmpi.b	#'.',d0
	beq	find_period
	dbra	d0,kkchs_lp
do_kkchs:
```


▶どんどん本がたまっていく。昔は「積ん読」する人というものが理解できませんでしたが、……なるほど理解できた気がします。　三森　浩一(23)東京都

```
    subq.l  #1,a0
    move.b  #'.',(a0)+
    move.b  #'Z',(a0)+
    move.b  #'M',(a0)+
    move.b  #'D',(a0)+
    clr.b   (a0)
    rts
find_period:
    cmpi.b  #' ',(a0)
    bls     do_kkchs                *'.'はあっても拡張子がないケース
    rts

main_ope:
    lea     sorce_name(pc),a0
    bsr     kakuchoshi

    lea     sorce_name(pc),a2
    bsr     fopen
    tst.b   d5
    bmi     open_err
    bsr     read                    *>d3.l=size a5.l=address

    move.l  (a5)+,d0
    cmpi.l  #$105a_6d75,d0  *ヘッダチェック
    bne     unid_err
    move.l  (a5)+,d0
    clr.b   d0
    cmpi.l  #$5369_4300,d0  *ヘッダチェック
    bne     unid_err
    lea     aru(pc),a0
    tst.b   (a5)
    bpl     prt_kekka
    lea     nai(pc),a0
prt_kekka:
    bsr     prt_sfn
    pea     (a0)
    DOS     _PRINT
    addq.w  #4,sp
    DOS     _EXIT

fopen:                              *ファイルのオープン(環境変数参照モード)
    * < a2=file name
    * > d5=file handle (error:d5<0)
    * - all
    movem.l d0/a0-a2,-(sp)

    clr.w   -(sp)
    pea     (a2)
    DOS     _OPEN
    addq.w  #6,sp
    move.l  d0,d5                   *d5.w=file handle
    bpl     exit_fopen              *no problem

    movea.l a2,a0                   *a0=file name only(non pathed)

    pea.l   env_work(pc)
    clr.l   -(sp)
    pea     getname(pc)
    DOS     _GETENV
    lea     12(sp),sp
    tst.l   d0
    bmi     exit_fopen

    lea.l   env_work(pc),a1
fopen_lp01:
    lea     open_fn(pc),a2
fopen_lp02:
    move.b  (a1)+,d0
    beq     do_fopen
    cmpi.b  #';',d0
    beq     do_fopen
    move.b  d0,(a2)+
    bra     fopen_lp02
do_fopen:
    moveq.l #'¥',d0
    cmp.b   -1(a2),d0
    beq     sva0
    cmp.b   (a0),d0
    beq     sva0_
    move.b  d0,(a2)+
sva0:
    cmp.b   (a0),d0                 *'¥'が続いた場合
    bne     sva0_
    addq.w  #1,a0
    bra     sva0
sva0_:
    pea     (a0)
do_fopenlp:
    move.b  (a0)+,(a2)+
    bne     do_fopenlp
    clr.b   (a2)
    move.l  (sp)+,a0

    clr.w   -(sp)
    pea     open_fn(pc)
    DOS     _OPEN
    addq.w  #6,sp
    move.l  d0,d5                   *d5.w=file handle
    bpl     exit_fopen              *no problem
    tst.b   -1(a1)                  *まだ環境変数が残ってるか
    bne     fopen_lp01
exit_fopen:
    movem.l (sp)+,d0/a0-a2
    rts

read:
    * < d5.l=file handle
    * > a5=data address
    * > d3.l=size
    * X d0
    move.w  #2,-(sp)                *ファイルの長さを調べる
    clr.l   -(sp)
    move.w  d5,-(sp)
    DOS     _SEEK
    addq.w  #8,sp                   *d0.l=file length
    move.l  d0,d3                   *d3=length
    beq     fsize0                  *file size=0
    bmi     fsize0                  *file size=minus

    move.l  d0,-(sp)
    DOS     _MALLOC
    addq.w  #4,sp
    tst.l   d0
    bmi     out_mem                 *OUT OF MEMORY
    move.l  d0,a5

    clr.w   -(sp)                   *ファイルポインタを元に戻す
    clr.l   -(sp)
    move.w  d5,-(sp)
    DOS     _SEEK
    addq.w  #8,sp

    move.l  d3,-(sp)                *push size
    pea     (a5)                    *push addr
    move.w  d5,-(sp)                *file handle
    DOS     _READ
    lea     10(sp),sp
    tst.l   d0
    bmi     read_err                *読み込み失敗

    move.w  d5,-(sp)                *close
    DOS     _CLOSE
    addq.l  #2,sp
    rts

print_title:
    move.w  #2,-(sp)
    pea     title(pc)
    DOS     _FPUTS
    addq.w  #6,sp
    rts

print_help:
    lea     help_mes(pc),a1
    bsr     mes_prt
bye_bye:
    DOS     _EXIT

err_bye:
    bsr     mes_prt
    move.w  #-1,-(sp)
    DOS     _EXIT2

prt_sfn:
    lea     sorce_name(pc),a1
    bra     mes_prt

read_err:
    bsr     prt_sfn
    lea     read_err_mes(pc),a1
    bra     err_bye

out_mem:
    lea     out_mem_mes(pc),a1
    bra     err_bye

open_err:
    bsr     prt_sfn
    lea     open_err_mes(pc),a1
    bra     err_bye

fsize0:
    lea     fsize0_mes(pc),a1
    bra     err_bye

unid_err:
    bsr     prt_sfn
    lea     unid_err_mes(pc),a1
    bra     err_bye

mes_prt:
    pea     (A1)
    DOS     _PRINT
    addq.w  #4,sp
    rts

work:
title:      dc.b $1b,'[35mCOMCHK.R',$1b,'[m version 1.00 (C) 1992 '
            dc.b $1b,'[36mZENJI SOFT',$1b,'[m',13,10,0
help_mes:   dc.b $1b,'[37m(使い方)',13,10
            dc.b $1b,'[m COMCHK <読み込みファイルネーム[.ZMD]>',13,10
            dc.b $1b,'[37m(機能)',13,10
            dc.b $1b,'[m'
            dc.b 'ZMDファイル中に共通コマンドがあるかチェックするよ。',13,10
            dc.b 'ZMSファイルに対して実行しても無意味。あしからず',13,10
            dc.b 0
read_err_mes:       dc.b 'が読めなかったよ',13,10,0
out_mem_mes:        dc.b 'メモリが不足しているよ! 増設しよう',13,10,0
open_err_mes:       dc.b 'が見つからなかったよ',13,10,0
fsize0_mes:         dc.b 'ファイルサイズが変だ!!',13,10,0
unid_err_mes:       dc.b 'はZMDデータじゃないみたいだね',13,10,0
aru:                dc.b 'には共通コマンドが存在するよ',13,10,0
nai:                dc.b 'には共通コマンドは無いね',13,10,0
getname:            dc.b    'zmusic',0
            .bss
sorce_name:         ds.b    91
open_fn:            ds.b    91
env_work:           ds.b    256
    .even
end_of_prog:
```

▶ この前，親戚の子供が遊びにきていたので，X68000で遊ばせてあげたところ「Z'sSTAFF」がいちばん気に入ったようでした。それにしても９月号の「ジャギー除去」のプログラムはすごいです。いままで描いた絵のほとんどに使ってしまいました。

山田　拓也(21)神奈川県

バックナンバー案内

BACK ISSUES

ここには1991年10月号から1992年9月号までをご紹介しました。現在1991年1，5，8，9，11，12，1992年1，2，4，5～9月号の在庫がございます。バックナンバーおよび定期購読の申し込み方法については，174ページを参照してください。

1991

10月号（品切れ）
特集　マシン語との邂逅

連載　響子 in CGわ～るど/マシン語プログラミング/ショートプロ/ハード工作/Z80's Bar/よいこのSX-WINDOW/ANOTHER CG WORLD/吾輩はX68000である/ようこそC言語/大人のためのX68000

●新連載　Computer Music入門

●NEW Print Shop PRO-68K Ver 2.0

LIVE in '91　うれしい! たのしい! 大好き!/SPANISH BLUE

THE SOFTOUCH　ボナンザブラザーズ/ロードス島戦記/ジーザスII他

全機種共通システム　Small-C活用講座（初級編）

11月号
特集　空間彷徨型ゲーム大分析

連載　響子 in CGわ～るど/大人のためのX68000/ANOTHER CG WORLD/DōGA/ショートプロ/Computer Music入門/吾輩はX68000である/ようこそC言語/マシン語プログラミング/Z80's Bar/ハード工作

●X68000用カードゲーム　ギャップ

●新製品紹介　F-Card GT

LIVE in '91　オーダイン

THE SOFTOUCH　キャメルトライ/アクアレス/フューチャーウォーズ他

全機種共通システム　Small-C活用講座（応用編）/MORTAL

12月号
特集　音・そして音楽とコンピュータ

別冊付録　X68000 THE GAME SOFTWARE BEST SELECTION

連載　響子 in CGわ～るど/マシン語プログラミング/ショートプロ/ハード工作/Z80's Bar/ようこそC言語/ANOTHER CG WORLD/吾輩はX68000である/Computer Music入門/大人のためのX68000

●エレクトロニクスショウ＆データショウ

LIVE in '91　OH YEAH!/サイレント・イヴ/ジングルベル

THE SOFTOUCH　フェアリーランドストーリー/プロサッカー68他

全機種共通システム　Small-C用　SLANGコンパチ関数

1992

1月号
特集　SX-WINDOWの未来

連載　響子 in CGわ～るど/DōGA・CGA/大人のためのX68000/ハード工作/Z80's Bar/ショートプロ/吾輩はX68000である/ANOTHER CG WORLD/Computer Music入門/カードゲーム

●MAGIC用ゲーム　3D MAZE

●CM-300/500&LA音源の活用法

LIVE in '92　DRAGON SABER/すき/THE ENTERTAINER

THE SOFTOUCH　出たな!! ツインビー/ブリッツクリーク/飛翔鮫他

全機種共通システム　パズルゲームLINER

2月号
特集　2Dグラフィックの拡張

連載　響子 in CGわ～るど/大人のためのX68000/マシン語プログラミング/ハード工作/ショートプロ/ANOTHER CG WORLD/Z80's Bar/吾輩はX68000である/Computer Music入門/カードゲーム

●TREND ANALYSIS

●MIRAGE MODEL STUFF/Press Conductor PRO-68K

LIVE in '92　ストリートファイターII/Tide Over

THE SOFTOUCH　ジェノサイド2/アルシャーク/コード・ゼロ他

全機種共通システム　シミュレーションゲームPOLANYI

3月号（品切れ）
特集　SCSIの活用

連載　響子 in CGわ～るど/DōGA・CGA/大人のためのX68000/Z80's Bar/ショートプロ/吾輩はX68000である/マシン語プログラミング/ハード工作/ANOTHER CG WORLD/Computer Music入門/カードゲーム

●Z-MUSIC支援ツール　ZPDCON.X

●Z's-EX用拡張コマンド　MASK_reverse.X

LIVE in '92　ギャラクシーフォース/君が代

THE SOFTOUCH　グラディウスII/レミングス/大戦略III'90/伊忍道

全機種共通システム　カードゲームKLONDIKE

4月号
特集　成熟するゲームと日本の文化

連載　よい子のSX-WINDOW/Z80's Bar/響子 in CGわ～るど/ショートプロ/吾輩はX68000である/ハード工作/ANOTHER CG WORLD/Computer Music入門

●発表　1991年度GAME OF THE YEAR

●バーコードバトラー

LIVE in '92　あじさいのうた/ショパン練習曲作品25-2へ短調/IT'S MAGIC

THE SOFTOUCH　ファーストクィーンII/マスターオブモンスターズII他

全機種共通システム　実践Small-C(1)オプティマイザO80

5月号
特集　明日のための環境づくり

第7回　言わせてくれなくちゃだワ

連載　響子 in CGわ～るど/大人のためのX68000/Z80's Bar/ハード工作/ショートプロ/マシン語プログラミング/Computer Music入門/吾輩はX68000である

●製品紹介　MIDI音源　03R/W/MIC68K

LIVE in '92　フレンズ/Danger Line

THE SOFTOUCH　エイリアンシンドローム/苦胃頭捕物帳他

全機種共通システム　実践Small-C(2)COMMAND.OBJ

6月号
特別企画　Oh!MZ,Oh!X10年間の歩み

特別付録　創刊10周年記念PRO-68K（5″2HD）

連載　響子 in CGわ～るど/大人のためのX68000/マシン語プログラミング/ハード工作/ショートプロ/ANOTHER CG WORLD/Z80's Bar/吾輩はX68000である/Computer Music入門

●新製品紹介　Z'sSTAFF PRO-68K ver.3.0

LIVE in '92　Shake the Street/Ancient relics

THE SOFTOUCH　スピンディジーII/ロイヤルブラッド/ライフ&デス他

全機種共通システム　実践Small-C講座(3)COMMAND.OBJ2

7月号
特集　超空間美術論

特別付録　DōGA CGAシステム＆お試しディスク（5″2HD）

連載　よいこのSX-WINDOW/響子 in CGわ～るど/Z80's Bar/ANOTHER CG WORLD/大人のためのX68000/Computer Music入門/ハード工作/ショートプロ

●試用レポート　V70アクセラレータボード

LIVE in '92　Bye Bye My Love/MATERIAL GIRL/ヴェクザシオン

THE SOFTOUCH　将棋聖天&棋太平68K/シムアース/太閤立志伝

全機種共通システム　実践Small-C講座(4)関数リファレンス

8月号
特集　プログラミング再入門

連載　響子 in CGわ～るど/吾輩はX68000である/よいこのSX-WINDOW/マシン語プログラミング/ハード工作/ANOTHER CG WORLD/大人のためのX68000/Computer Music入門/ショートプロ

●新製品紹介　MATIER/TG100/SOUND SX-68K

LIVE in '92　氷穴/ガラガラヘビがやってくる/風の贈り物

THE SOFTOUCH　三國志III/シムアース/ウルティマVI/バトルテック

全機種共通システム　実践Small-C講座(5)ワイルドカード
グラフィックライブラリGRAPH.LIB

9月号
特集　数値演算の熱い逆襲

連載　DōGA CGアニメーション講座/大人のためのX68000/響子 in CGわ～るど/吾輩はX68000である/ショートプロ/マシン語プログラミング/ハード工作/ANOTHER CG WORLD

●新製品紹介　MATIER/MIREGE System Model Stuff

LIVE in '92　恋をしようよ Yeah! Yeah!/ゆめいっぱい

THE SOFTOUCH　ファイナルファイト/ライジングサン/ヨーロッパ戦線/シューティング68K GAMES

全機種共通システム　O-EDIT & MODCNV

よいこのSX-WINDOW講座 ————————(第10回)

スクロールバーに挑戦

お待たせしました。今回は，以前予告したままになっていたスクロールバーを取り上げます。要はボリューム型のコントロールと同じなのですが，画面の書き換え処理などを同時に行わなければなりません。サンプルを参考にしてみてください。

Nakamori Akira 中森 章

はじめに

今回の予定はタスクマンの関数を使ってタスク間通信を行う予定でしたが，ちょっと気が変わりました。この連載の最初のほう（2回目だか3回目）で，次回はスクロールバーについて説明するといって，そのままにしていたのを思い出しました。もし，スクロールバーの説明を心待ちにしていた人がいたらごめんなさい。

スクロールバーに関しては私自身頭の中にもやもやとしたものがあったので，知らず知らずのうちに避けていたのかもしれません。しかし，最近ではそのもやもやも少しは晴れてきています。近くSX-WINDOW開発キットも発売になるので，このまま放っておいてもいいかなと思ったのですが，開発キットの途中版のスクロールバーの説明はあまり的を射ていないような気がしました。と，いうわけで今回はスクロールバーの説明をしたいと思います。

スクロールバーとは

そもそもスクロールとはエアコンの売りのひとつ，じゃなくて，巻物のことです。巻物とは，いうまでもなく，細長い紙に絵や文章を書いて軸に巻いたものです。それを見るときには，巻いた紙をほぐしながら，そこから次々と現れてくる部分を見ていきます（図1）。

その見える部分だけに注目すると，目の前を絵や文字が左から右へと移動していくような感じを受けます（これは巻物が縦書きで，右から左に文章が書かれていると仮定した場合)。

このように見えている部分がずりずりと移動していき，代わりに隠れていた部分が次々と見えてくるような効果を与えることをスクロールするといいます。

スクロールバーとは，ものを(強制的に)スクロールさせるための指示を与えるものです。多くの場合，スクロールバーをマウスでクリックしたり，その一部をドラッグすることにより，画面に表示されている図形や文字をスクロールさせることができます。

SX-WINDOWにおいて，スクロールバーは標準ボタンやアップダウンボタンやスライドボリュームなどの制御ボタン（コントロール）の一種として扱われます。スクロールバーは画面がどの程度スクロールしているのかを，その値として持っています。通常は値が0ならば初期状態で，値が大きくなるほど初期状態から大きくスクロールしている，というように意味づけることが多いようです。

スクロールバーの値を変更したり位置や大きさを変更することはコントロールマンの関数によって行います。ただし，この値を変更するという処理は実際に画面をスクロールする処理とは別物です。スクロールバーの値を変更したからといって，画面が自動的にスクロールするわけではありません。スクロールバーを使用するプログラムでは，スクロールバーの値に応じて画面をスクロールするという処理が別途必要になります。この場合，スクロールバーの値と画面のスクロールを関連づけるのはユーザープログラムの責任です。

とりあえず，ここでは，SX-WINDOWにおけるスクロールバーの各部の名称と，それらの部分がマウスで選択されたときに行うべき代表的な処理について示しておきましょう。

図2にスクロールバーの各部がマウスで選択されたときにプログラムに通知されるパートコードについて示します。SX-WINDOWのスクロールバーでは中心にあるアップボタンとダウンボタンのある四角の領域がスクロールされる領域を示しています。これはスクロールバーを操作することで見ることのできる最大の領域（デスティネーションレクタングル）のうち，現在ウィンドウ上に見えている領域（ビューレクタングル）に対応しています[1]。

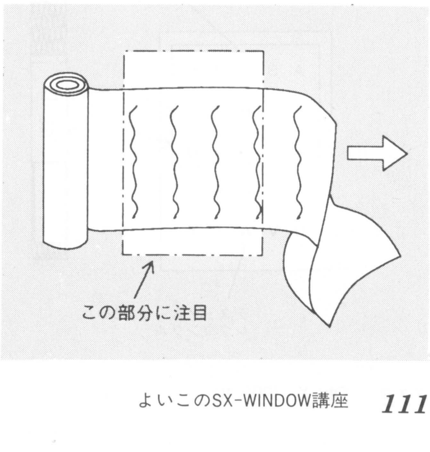
そして，その後ろにあるサムの部分が，ビューレクタングルに対するデスティネーションレクタングルの位置のずれ具合を示しています。つまり，サムが上方（左方）にあればデスティネーションレクタングルの下の部分がビューレクタングルに見えていることになり，サムが下方（右方）にあればデスティネーションレクタングルの上の部分が見えていることになります(図3)。

それでは，それぞれのパートコードがマウスで選択（その位置にマウスカーソルを持っていき左ボタンを押すこと）されたときに，どのような処理をすべきかを説明します。

●C_INDOWN（ダウンボタン）

デスティネーションレクタングルの下方

図1　スクロールは巻物



（右方）が見えるように，ビューレクタングルに対してデスティネーションレクタングルを上方（左方）に移動させる処理を行います。

スクロールバーの値は増加させます。また，このときサムは上方（左方）へ移動します。

●**C_INUP**（アップボタン）

デスティネーションレクタングルの上方(左方)が見えるように，ビューレクタングルに対してデスティネーションレクタングルを下方（右方）に移動させる処理を行います。

スクロールバーの値は減少させます。また，このときサムは下方（右方）へ移動します。

●**C_INPGUP**（スクロール領域・アップ側）

マウスカーソルで指定する位置まで直接サムを移動します。ビューレクタングルに対してデスティネーションレクタングルを下方（右方）に移動させる処理を行います。

スクロールバーの値は減少させます。また，このときサムは下方（右方）へ移動します。これはマウスカーソルがある方向です。

C_INUPとの違いはスクロールバーの値の変化がマウスカーソルの位置によって決定されることです。C_INUPではある一定値だけ変化させます。

●**C_INPGDOWN**（スクロール領域・ダウン側）

マウスカーソルで指定する位置まで直接サムを移動します。ビューレクタングルに対してデスティネーションレクタングルを上方（左方）に移動させる処理を行います。

スクロールバーの値は増加させます。また，このときサムは上方（左方）へ移動します。これはマウスカーソルがある方向です。

C_INDOWNとの違いはスクロールバーの値の変化がマウスカーソルの位置によって決定されることです。C_INDOWNPではある一定値だけ変化させます。

●**C_INTHUMB**（サム，またはスクロールボックス）

サムはマウスの左ボタンを押しながらドラッグしてスクロールバーの好きな位置まで移動させます。その位置に合うように，ビューレクタングルに対してデスティネーションレクタングルを上下（左右）に移動させる処理をします。

スクロールバーの値もサムの位置に対応して増減させます。

C_INPGUP，C_INPGDOWNとの違いは，スクロールする度合いをきめ細かく指定できる点でしょう。

図2　スクロールバーとパートコード

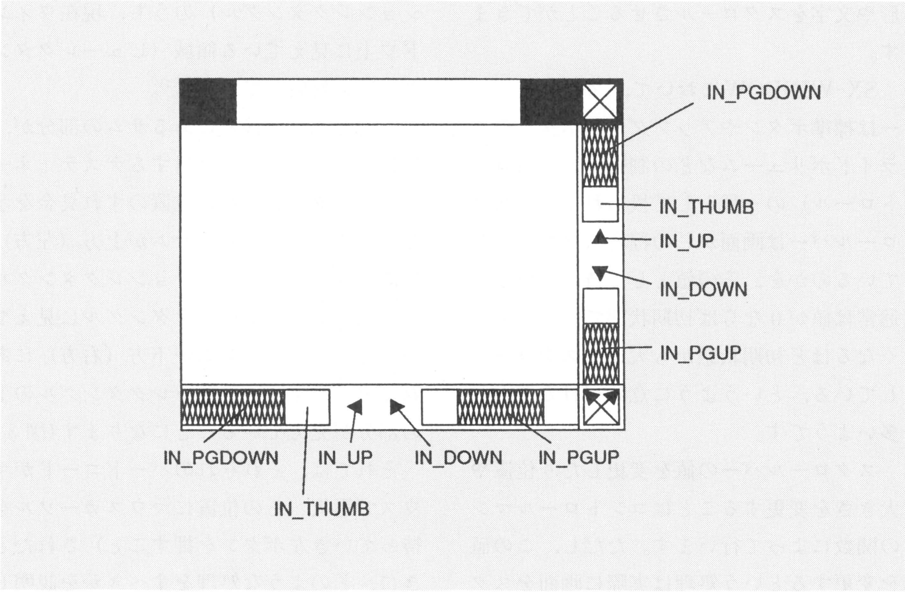
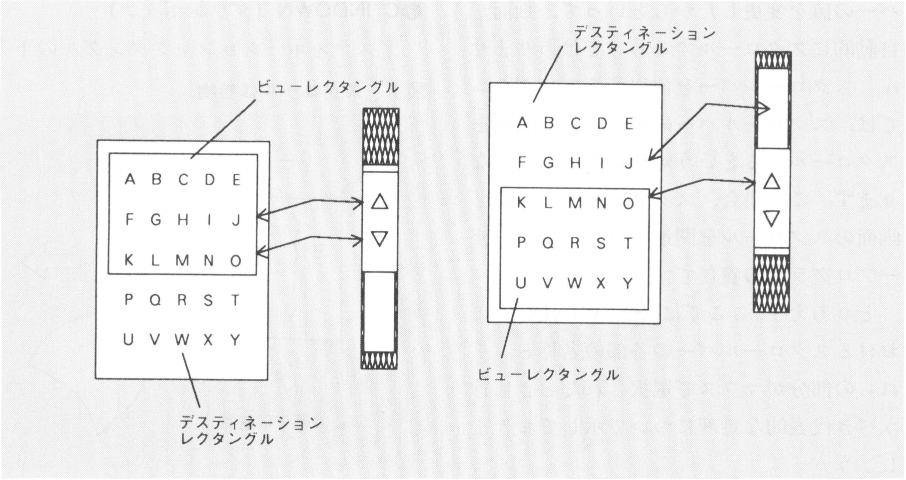


図3　スクロールバーとサムの位置関係



1) デスティネーションレクタングル，ビューレクタングルというのはテキストマンで扱うテキストのスクロールで使用する用語。ここでいうデスティネーションレクタングルはビットマップレクタングル，ビューレクタングルはグラフポートレクタングルのこと。ウィンドウのスクロール処理をテキストのスクロール処理と同一視するために，あえてそう呼んでいる。

画面をスクロールする方法

スクロールバーの値に応じて画面をスクロールする方法には，大きく分けて２種類の方法があります。それらについて説明しましょう。

●**描画側で辻褄を合わせる方法**

これはウィンドウに文字や図形を描画するとき，スクロールバーの値に従って，文字や図形をずらして描画する方法です。たとえば，デスティネーションレクタングルが，

ABCDEFGHIJK

で，ビューレクタングルが３文字分を表示できる，

のような領域であるとします。このとき，

初期状態（スクロールしていない状態）では，

ABC　DEFGHIJK

のように，デスティネーションレクタングルの左３文字がビューレクタングルから見えています。

もし，スクロールバーの値がnであるときは，デスティネーションレクタングルをn文字だけずらして描画するのがこの方法です。スクロールバーの値が２ならば，デスティネーションレクタングルを２文字ずらして描画します。つまり，ウィンドウにはCの文字から描画します。そうすれば，

CDE　FGHIJK

のようになり，ビューレクタングルの内容はスクロールしたように見えます。

この方法の利点は容易に実現できることです。描画する位置をずらすだけでよいので，あとで述べるもうひとつの方法と異なり，ビューレクタングルやスクロールバーを移動する処理が不要になります。

欠点は描画する側で，描画とは直接関係ないはずの，スクロールバーの値を意識しなければならないことでしょう。SX-WINDOWのプログラムはスケルトンと呼ばれる骨格プログラムを作っておいて，それを基に作成することがほとんどです。しかし，この方法では，スクロール処理が個々の描画ルーチンに依存するため，スケルトンにスクロール処理を盛り込んでおくことはできません。

●描画側にはスクロールを意識させない方法

これはスクロールバーの値に応じてビューレクタングルのほうを移動させる方法です。前と同様に，デスティネーションレクタングルが，

ABCDEFGHIJK

で，ビューレクタングルが３文字分を表示できる，

であるとします。このとき，初期状態（スクロールしていない状態）では，

ABC　DEFGHIJK

のように，デスティネーションレクタングルの左３文字がビューレクタングルから見えています。スクロールバーの値がnであるとき，ビューレクタングルをn文字だけ移動するのがこの方法です。スクロールバーの値が２ならば，ビューレクタングルを２文字ずらして描画します。そうすれば，

AB　CDE　FGHIJK

のようになって，ビューレクタングルから見える文字はスクロールして見えます。

この方法を利用する場合，実際にビューレクタングルを移動させるわけではありません。（ローカル）座標系の原点を移動して，ビューレクタングルの存在する（ローカル）座標をずらします。デスティネーションレクタングルには，常に同じ（ローカル）座標で示される位置から文字や図形を描画することにすれば，ビューレクタングルから見える文字や図形はスクロールして見えるのです。描画する側はスクロールバーの値を意識する必要はありません。

このため，ウィンドウのスクロール処理の手順を一般化することができ，スケルトンなどの基本的なプログラムにスクロールバーの処理を盛り込むことができるようになります。そのようなスケルトンを用いるアプリケーションでは，スクロールバーの存在など意識せず，普通にプログラムを作ればよいのです。極端な話，（スケルトンに対して追加する部分の）プログラムでなにも考えなくても，スクロールバーで画面をスクロールできるのです。

しかし，この処理の欠点はスケルトンのプログラムが非常に複雑になる点です。座標系の基準位置をずらすのですから，スクロール処理を行うたびにローカル座標で管理されているすべてのコントロール（スクロールバーをはじめ，標準ボタンやスライドボリュームなど）の位置がずれてしまいます。したがって，スクロールバーの値に応じてコントロールを移動させる処理をしなければなりません。

スクロールバーの処理

スクロールバーの値に従って描画位置を変える方法は誰でも考えつく方法です。ここで，取り立てて説明するほどのことはないでしょう。以後は，少し複雑な処理である，座標系をずらしてスクロールを行う処理について説明することにします。

なお，スクロール処理には，ウィンドウ内の一部の領域をスクロールする場合と，ウィンドウ内の全体をスクロールする場合の２種類があります。どちらも，基本的な処理は同じですが，以後はウィンドウ内の全体をスクロールさせるものとして話を進めていきます。

●スクロールバーの値

まずは，スクロールバーの持つ値です。処理を一般化するためには，値の意味づけはプログラムに任せられているとはいえ，すべての場合で一意に定まる値を持たせなければなりません。これは，デスティネーションレクタングルに対するビューレクタングルの移動量をそのままスクロールバーの値とすればよいでしょう。つまり，横スクロールバーなら横方向のX座標のずれ，縦スクロールバーなら縦方向のY座標のずれを値とします。

特殊な場合を除き，ビューレクタングルはウィンドウの内部と一致しています。スクロールを行うことは，ビューレクタングルのローカル座標をずらすことです。初期状態ではウィンドウ内部の左上のローカル座標は(0,0)ですから，この座標がスクロールによってずれることになります。ウィンドウの左上のローカル座標が，それぞれ，横スクロールバーと縦スクロールバーの値を示しています（図４）。

●スクロールバーの最大値と最小値

デスティネーションレクタングルは（スクロールバーの値を変化させることにより）ビューレクタングルをずらしていって見ることのできる最大の範囲です。したがって，スクロールバーの値にはビューレクタングルとデスティネーションレクタングルの大きさに依存した最大値と最小値が存在します。図５を見ればわかるように，最小値は０，最大値は横スクロールバーは，

デスティネーションレクタングルの横

幅－ビューレクタングルの横幅
縦スクロールバーが，

デスティネーションレクタングルの縦幅－ビューレクタングルの縦幅

となります。したがって，ウィンドウの大きさ（ビューレクタングル）が変化した場合は，スクロールバーの最大値と最小値を変更しなければなりません（さすがにデスティネーションレクタングルの大きさは不変でしょう）。最小値は0に決まっているので，実際には最大値のみの変更になります。

●画面のスクロール処理

スクロールの程度（スクロール量＝スクロールバーの値）が，座標のずれを表すことにすれば，スクロール処理は簡単です。ウィンドウ内部の左上の座標を，スクロールバーの値に応じて付け替えてやればいいのです。これは，

GMSetHome

関数で行います。つまり，X座標のずれ（横スクロールバーの値）とY座標のずれ（縦スクロールバーの値）を組にしたpoint_t型のデータを引数にしてGMSetHome関数を呼び出します。これでスクロール処理はおしまいです。

とはいえ，これでは座標系をずらしただけで，次にアップデートイベントが発生するまで，画面の再描画は行われません。このため，強制的に再描画を行う処理が必要です。

図4　左上の座標とスクロールの関係

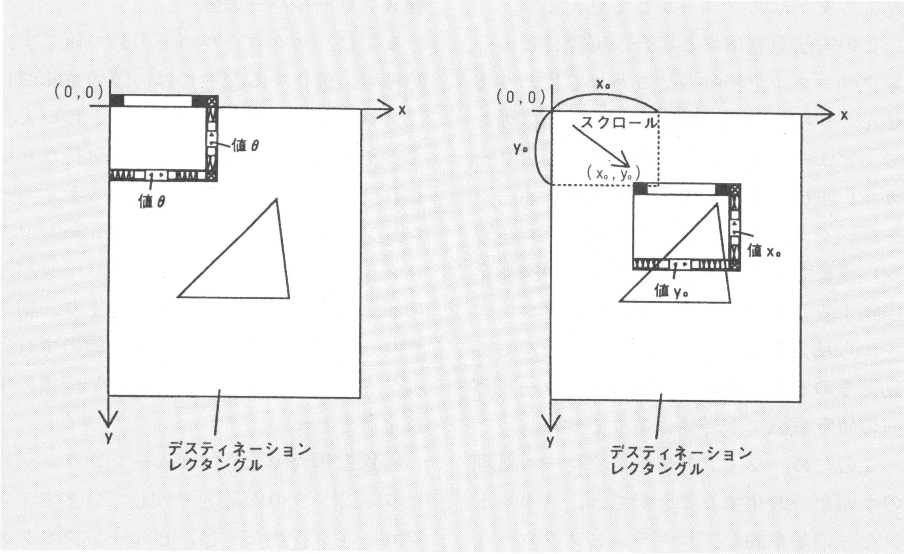
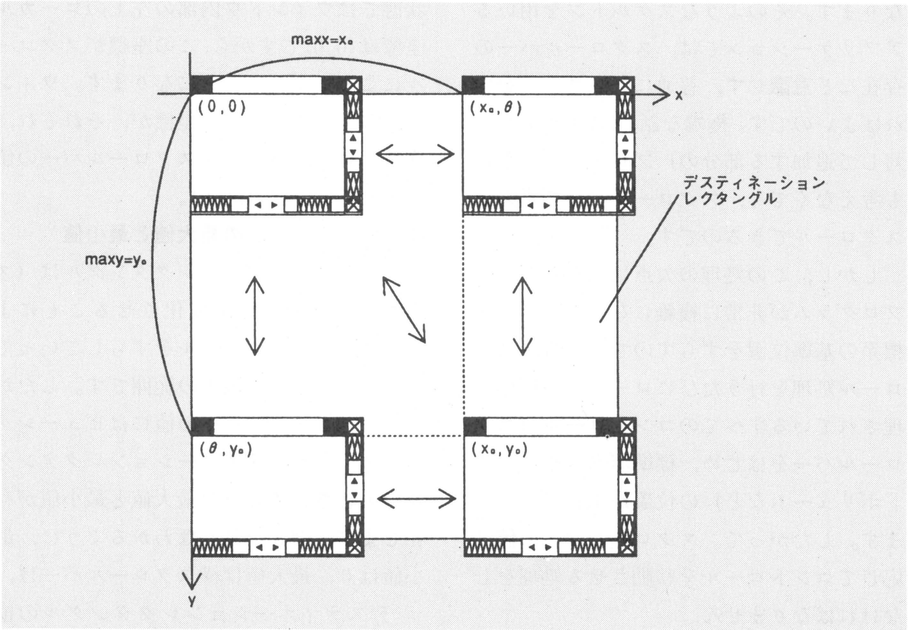


図5　スクロールバーの値が取る範囲



再描画を行う場合，単純には，画面（ウィンドウ内部）を一度消去することを考えつきます（そうでなければ新しい文字が前の文字と重なってしまう）。しかし，この方法ではスクロールするごとに画面がちらついてしまい，スクロールを行ったという感じが出ません（いきなり画面が変化したように見える）。そこで，スクロールを行う場合は，画面を消去する代わりに，

GMScroll

関数を用いて，座標をずらすのと同じ量だけ画面の表示内容をずらし，その後，実際に座標系をずらしてから再描画を行います。これにより，スムーズにスクロールした感じを出すことができます（図6）。

GMScroll関数自身は，座標系をずらすわけでもなんでもなく，指定した領域内に表示されている図形をスクロールして表示するだけですから，そのあとに座標系をずらすことが必要になるのです。

●スクロールバーの移動

GMSetHome関数はビューレクタングル（グラフポートレクタングル）を移動します。このため，ローカル座標で管理されている制御ボタン類の位置が移動してしまいます。このとき，ウィンドウの右端と下側に付いているスクロールバーの位置がずれてしまいますから，GMSetHome関数を呼び出した場合は，スクロールバーの位置をウィンドウ内部に収まるように移動しなければなりません（図7）。

スクロールバーを移動・拡大するときには，スクロールバーをいったん消去してから，移動・拡大を行い，そして再び表示するというのが常識です。スクロールバーを消去する理由は，移動する前の位置にスクロールバーの表示が残ってしまうのを防ぐためです。

しかし，スクロールバーの（拡大を伴わない）移動では，垂直方向の移動か水平方向の移動しかないので，前の位置にあるスクロールバーは新しい位置のスクロールバーによって完全に隠されてしまうため，わざわざスクロールバーを消去する必要はありません。消去を行うと画面がちらついて，かえって見にくくなってしまいます。

●ウィンドウのズームイン，ズームアウト

ここでは，ウィンドウのスクロールをウィンドウ内部の左上の座標を変更すること

で実現しています。ところが，ウィンドウをズームイン（グローボックスのダブルクリックによる縮小），あるいはズームアウト（グローボックスのダブルクリックによる拡大）する場合には，ウィンドウの四隅の座標が勝手に変更されてしまいます。

このとき，ウィンドウ内部の左上の座標は(0,0)になります。もし，それまでがスクロールを行った状態にあると，ウィンドウの左上の座標が勝手に変化してしまうために，スクロールバーの値と辻褄が合わなくなってしまいます。これを避けるひとつの方法はズームアウトやズームインが行われる場合は，スクロールバーの値を強制的に0にしてやることです。

しかし，あとで示すサンプルプログラムでは，ズームインやズームアウトが発生しそうな場合（マウスの左ボタンが押された場合）はウィンドウの左上の座標を覚えておき，ズームアウトやズームインが発生したあとにウィンドウの左上の座標を以前の値に戻してやるという方法をとっています。

●タスクマンを使用する方法

ウィンドウの移動や拡大処理はタスクマンの関数であるSXCallWindM，またはSXCallWindM2によってタスクマンに任せてしまうことができます。制御ボタンも同じく，タスクマンの関数であるSXCallCtrlMでタスクマンに任せることができます。SXCallCtrlM関数はウィンドウのスクロールバーの処理までも自動的に行ってくれるので，この関数を利用すれば，いままで説明した処理をなにも考えなくてすみます。

ただし，SXCallCtrlMはスクロール範囲をウィンドウ内部全体，かつ，横方向，縦方向のスクロールバーが両方存在することを仮定していますから，スクロール処理の自由度が落ちてしまいます。とはいえ，それ以外の使用法のほうが希という気がしますから，プログラムの行数を節約するためにはSXCallCtrlM関数を利用したほうが得策でしょう。

SXCallCtrlM関数を利用する場合でも，ウィンドウの大きさが変わるような処理を行うときは，スクロールバーの最大値と最小値を変更することを忘れてはなりません。

以上が，ウィンドウ内部のスクロール処理の大まかな説明です。処理の手順をはっきりさせるために，図8にスクロール処理のフローチャートを示しておきましょう。

プログラムの例

図8に示した処理手順に従ってウィンドウ内部をスクロールさせるためのサンプルプログラムをリスト1に示します。リスト

図6　GMscrollの処理

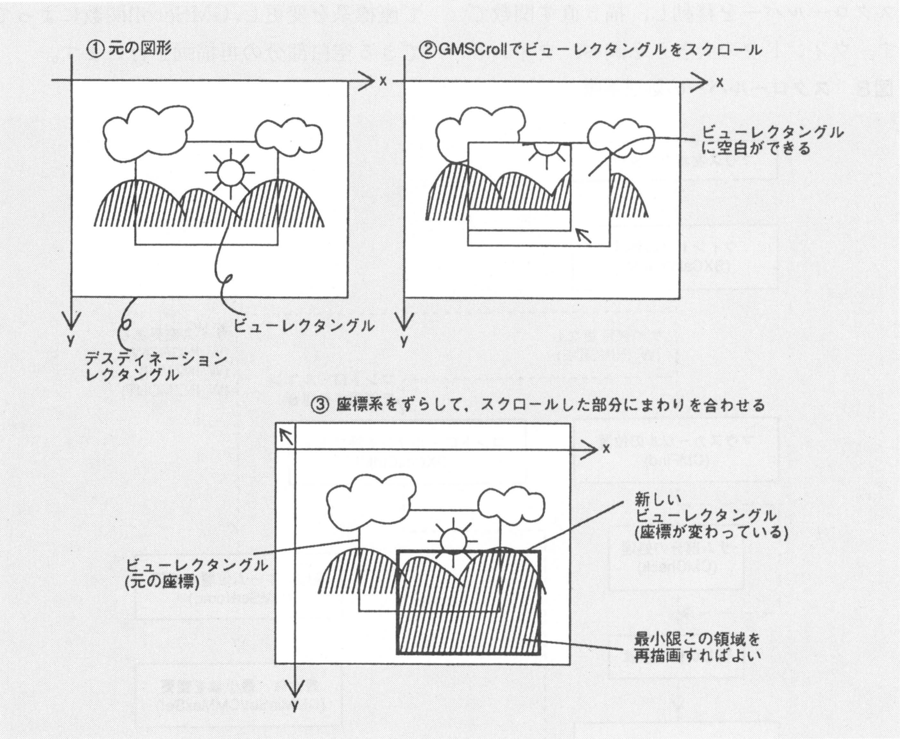
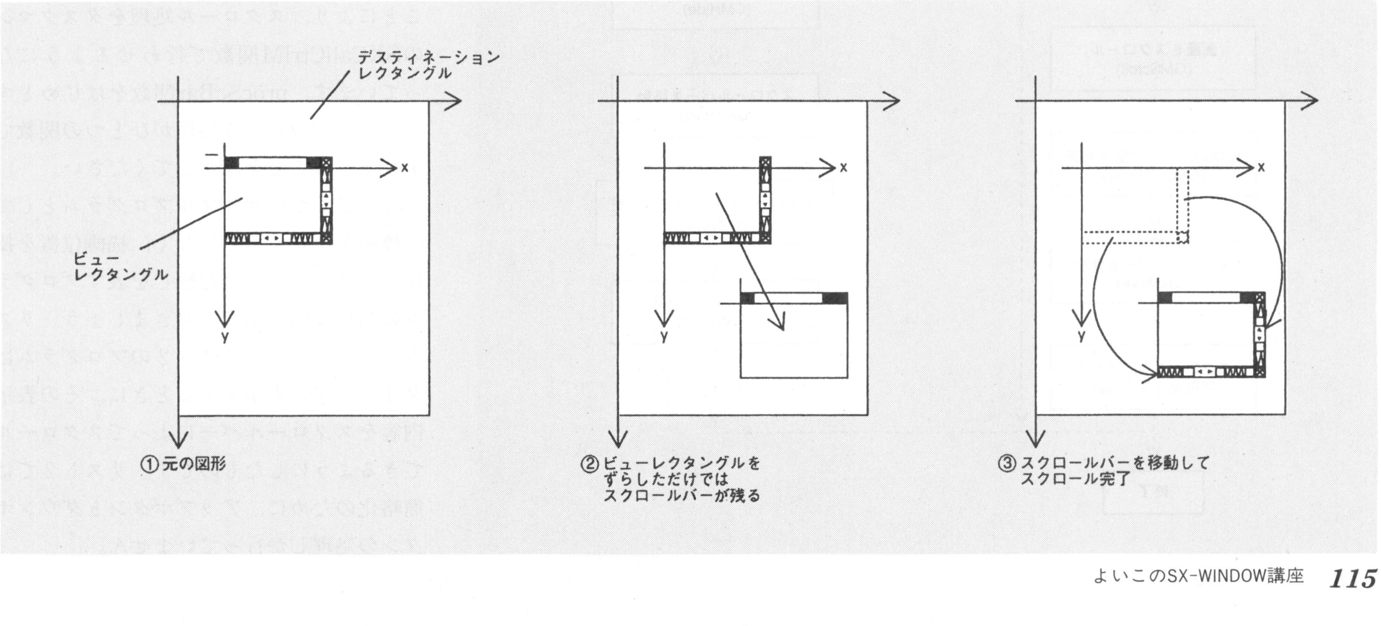


図7　スクロールバーの移動



1の実行結果を写真1に示します。

リスト1では，スクロールバーに関するいろいろな処理を機能別に関数に分けて行っています。それぞれの関数について，その処理内容を説明しておきましょう。

●MkScBar()

これは，スクロールバーのハンドルを定義する関数です。ウィンドウの右端と下側にスクロールバーを生成し，それぞれのハンドルをscBarV，scBarHという変数に格納します。

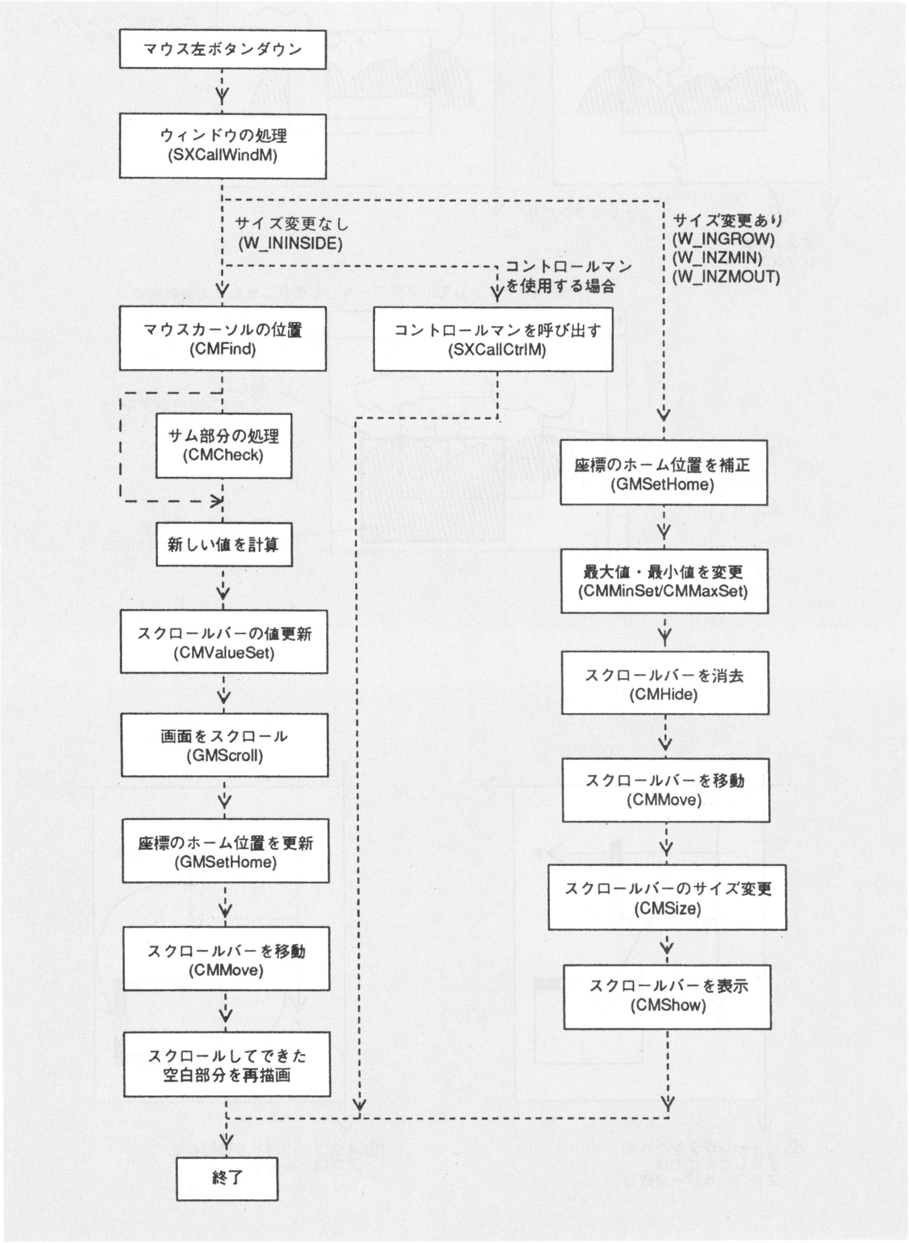
●UpdScBar (int resize)

これは，ウィンドウの大きさに合わせてスクロールバーを移動し，描き直す関数です。ウィンドウの大きさを調べ，ウィンドウの右端と下側にスクロールバーを描き直します。resizeという引数が0ならばスクロールバーを消去せずに描き直します。resizeが0以外なら，スクロールバーを消去してから描き直します。ウィンドウの大きさが変わる(可能性のある)場合はresizeを1にして呼び出し，そうでない場合はresizeを0にして呼び出します。

●DoScroll (point_t pt)

これは，ウィンドウ内のスクロールを行う関数です。ptにX方向とY方向の移動量を入れて呼び出します。GMScroll関数で表示内容をスクロールし，GMSetHome関数で座標系を変更し，GMScroll関数によってできる空白部分の再描画を行います。

図8　スクロールバーの処理手順



●NewScValue(control ＊＊selHdl, point_t pt)

この関数は，スクロールバーのIN_PGUP，IN_PGDOWNのパートがマウスでクリックされたときに，その位置がスクロールバーの最大値に対してどの程度の位置であるかを計算します。それは，次に設定するべきスクロールバーの値です。selHdlが垂直スクロールバー，あるいは水平スクロールバーのハンドルで，ptはマウスカーソルのローカル座標です。この関数は，要するに，

　　マウスのローカル座標/スクロールバーの最大値

の値を計算して返します。

●ChangeMaxValue()

これは，スクロールバーの最大値を設定し直すための関数です。ウィンドウの大きさが変化したときに呼び出します。

●procScBar(int part,control ＊＊selHdl,point_t pt)

この関数はスクロール処理のためのメインルーチンです。この関数がこれまでに説明してきた関数を呼び出してウィンドウのスクロール処理を実現します。

マウスがクリックされたときのパートコードであるpart，スクロールバーのハンドルであるselHdl，マウスカーソルのローカル座標を引数にして呼び出します。

なお，リスト1ではコンパイル時に，

　　－DCTRL_MAN

　(XCでは/DCTRL_MANでも可)

というコンパイルオプションを指定して，CTRL_MANというシンボルを定義することにより，スクロール処理をタスクマンのSXCallCtrlM関数で行わせるようになっています。procScBar関数をはじめとする多くの関数による処理がひとつの関数で実現できることを確認してください。

もうひとつのサンプルプログラムとして，座標系をずらすのではなく，描画位置を操作してスクロールした感じを表すプログラムの簡単な例を示しておきましょう。リスト2がそれです。リスト2のプログラムはダイアログの表示をするときに，その表示内容をスクロールバーによってスクロールできるようにしたものです。リスト2では簡略化のために，アップボタンとダウンボタンの処理しか行っていません。

リスト２のプログラムは別のプログラム（たとえばリスト１）からdoDialogという関数を呼び出して実行します。リスト１のプログラムと結合するためには，

−DMENU

（XCなら，あるいは/DMENU）

というコンパイルオプションを指定して，リスト１と一緒にコンパイルしてください。そうすれば，マウスの右ボタンをクリックすることでポップアップメニューが現れ，その中の「ものろおぐ」という項目を選択すると，リスト２のプログラムが実行されるようになっています。リスト２のプログラムの実行結果を写真２に示します。

おわりに

長年，構想を温めてきた（？）スクロールバーに関して初めて説明してみました。SX-WINDOW開発キットが発売になるようですので，それに負けるものかと先駆けての説明です（この記事が掲載される頃に開発キットが発売されていたら笑ってやってください）。私としては，これで溜まっていたものが一気に吐き出た感じで，すっきりしています。ちょっと説明をはしょった感もありますが，ひととおりの処理手順は示したつもりです。スクロール処理に関しては，説明した以上に細かい処理が必要になりますが，それらについてはリストのコメントを参考にしてください。本筋を理解していれば，あとは辻褄合わせのようなものですから説明は省いています。

さて，次回は今月やるつもりだったタスク間通信について説明したいと思っています。それでは，次回までさようなら。

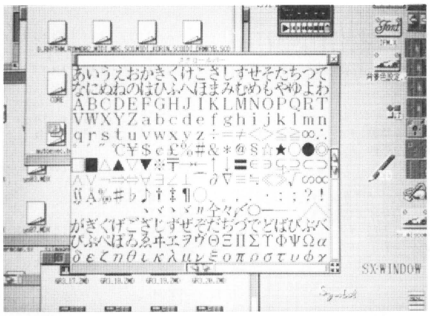


写真１

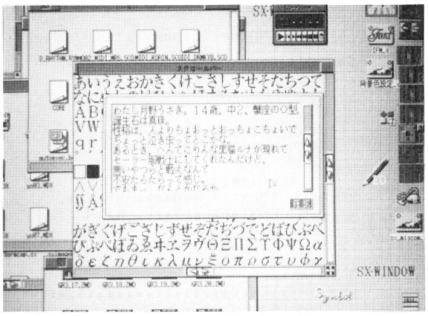


写真２

＜参考文献＞

1)　吉沢正敏，SX-WINDOWプロラミング，ソフトバンク，1991年.

2)　吉沢正敏，追補版SX-WINDOWプロラミング，ソフトバンク，1991年.

リスト１

```
===============  SX1  ==================
/*

     SX-WINDOW  スクロールバーのサンプル

                              (C) 中森 章, Aug.01, 1992
*/
#include <stdio.h>
#define __POINT_T   /* point_t 型を使う */
#include <sxlib.h>
#define FALSE   0
#define TRUE    ~FALSE
#define POINT(x,y)  (((x&0xffff)<<16)|(y&0xffff))
/*
     ここでウィンドウに関する定数を設定
*/
#define WDEFID       49
#define WINOPT       ( WC_GBOX | WC_GBOXON )
#define WINWIDTH     (14*24+18)
#define WINHIGHT     (10*24+18)
#define WINTITLE     "¥016スクロールバー"
#define EVENTMASK    EM_EVERY
/*
     スクロールバー用の定数
*/
#define SCBAR_WIDTH 18  /* スクロールバーの横幅 */
#define DEST_WIDTH  600 /* dest rectangle の横幅 */
#define DEST_HIGHT  384 /* dest rectangle の縦幅 */
/*
     ここは定数から計算される定数
*/
#define WINOPTL      ( WINOPT & 0xf )
#define WINDEFID     ( WDEFID << 4 | WINOPTL )

window  *winPtr;
rect    winSize;
event   eventRec;
int activeFlag;
int ctrlFlag;   /* コントロールがあるかないか */

control **scBarV;   /* 垂直スクロールバー */
control **scBarH;   /* 水平スクロールバー */
int     maxLimV;    /* 垂直バーの最大値リミット */
int     maxLimH;    /* 水平バーの最小値リミット */

#ifdef __GNUC__
asm( ".xdef _STACK_SIZE"  );
asm( "_STACK_SIZE equ   8192" );
asm( ".xdef _HEAP_SIZE"   );
asm( "_HEAP_SIZE equ    16384" );
#endif
/**************************************
*    スクロールバーを定義する         *
**************************************/
MkScBar()
{
     rect    cr;
     int w_width,w_hight;

     w_width = winPtr->wGraph.grRect.right  - winPtr->wGraph.
grRect.left;
     w_hight = winPtr->wGraph.grRect.bottom - winPtr->wGraph.
grRect.top;

     cr.left   = winPtr->wGraph.grRect.right - SCBAR_WIDTH;
     cr.top    = winPtr->wGraph.grRect.top;
     cr.right  = winPtr->wGraph.grRect.right;
     cr.bottom = winPtr->wGraph.grRect.bottom - SCBAR_WIDTH;

     maxLimV = DEST_HIGHT-(cr.bottom-cr.top);   /* 垂直スクロール
バーの最大値 */

     scBarV=CMOpen(winPtr,&cr,(LASCII*)"",-1,0,0,maxLimV,CI_S
CLBRWV<<4,0);

     cr.left   = winPtr->wGraph.grRect.left;
     cr.top    = winPtr->wGraph.grRect.bottom - SCBAR_WIDTH;
     cr.right  = winPtr->wGraph.grRect.right - SCBAR_WIDTH;
     cr.bottom = winPtr->wGraph.grRect.bottom;

     maxLimH = DEST_WIDTH-(cr.right-cr.left);   /* 水平スクロール
バーの最大値 */

     scBarH=CMOpen(winPtr,&cr,(LASCII*)"",-1,0,0,maxLimH,CI_S
CLBRWH<<4,0);

     CMDraw(winPtr);
}
/**************************************
*    ウィンドウの大きさに合わせて     *
*    スクロールバーを描き直す         *
**************************************/
UpdScBar(resize)
int resize;
{
     rect    r;
     int newV,newH;

     point_t xyV,xyH,sizV,sizH;

     GMSetGraph( &winPtr->wGraph );

     r = winPtr->wGraph.grRect;

     xyV.p.x = r.right-SCBAR_WIDTH;  /* 垂直バーの新しい位置 (X) */
     xyV.p.y = r.top;                /* 垂直バーの新しい位置 (Y) */
     xyH.p.x = r.left;               /* 水平バーの新しい位置 (X) */
     xyH.p.y = r.bottom-SCBAR_WIDTH; /* 水平バーの新しい位置 (Y) */

     if(resize){
          sizV.p.x = SCBAR_WIDTH;      /* 垂直バーの新しいサイズ (X方向
) */
          sizV.p.y = r.bottom-r.top-SCBAR_WIDTH;
                                       /* 垂直バーの新しいサイズ (Y方向
) */
          sizH.p.x = r.right-r.left-SCBAR_WIDTH;
                                       /* 水平バーの新しいサイズ (X方向
) */
          sizH.p.y = SCBAR_WIDTH;      /* 水平バーの新しいサイズ (Y方向
) */

```

```
            CMHide( scBarV );                   /* サイズ変更のため垂直バーを消す
*/
            CMHide( scBarH );                   /* サイズ変更のため水平バーを消す
*/
        }

        CMMove( scBarV, xyV );              /* 垂直バーを移動 */
        CMMove( scBarH, xyH );              /* 水平バーを移動 */

        if(resize){
            CMSize( scBarV, sizV );         /* 垂直バーのサイズ変更 */
            CMSize( scBarH, sizH );         /* 水平バーのサイズ変更 */
            CMShow( scBarV );               /* 垂直バーを表示 */
            CMShow( scBarH );               /* 水平バーを表示 */
        }
}
/*****************************************
*       ウィンドウコンテンツの              *
*       スクロール処理をする                *
*****************************************/
DoScroll(pt)
point_t pt;
{
        region  **updRgn;
        rect    clip;
        point_t p0,p1;

        clip=winPtr->wGraph.grRect;
        clip.right  -= SCBAR_WIDTH; /* スクロールする範囲を計算 */
        clip.bottom -= SCBAR_WIDTH;

        updRgn=GMNewRgn();          /* GMScroll 用のリージョンを定義 */

        p1.p.x = -pt.p.x;           /* スクロール用オフセット */
        p1.p.y = -pt.p.y;

        GMScroll( &clip, p1, updRgn );  /* スクロールする */

        GMSlideRect( &clip, pt );    /* 表示範囲も変わるので    */
        p0.p.x=clip.left;            /* 新しいホーム位置を計算 */
        p0.p.y=clip.top;
        GMSetHome( p0 );             /* ホーム位置を変更 */

        UpdScBar(0);                 /* スクロールバーを移動 */

        p1.x_y=0;                    /* ローカル座標の原点 */
        p1.x_y=GMLocalToGlobal( p1 );
        GMSlideRgn( updRgn, p1 );    /* グローバル座標に変換 */
        GMSlideRgn( updRgn, pt );    /* スクロールした分移動 */
        WMAddRgn( updRgn );          /* アップデートリージョンに追加 */

        WMUpdate( winPtr );          /* 描き直し */
        DRAW();
        WMUpdtOver( winPtr );
        GMDisposeRgn( updRgn );

}
/*******************************************
*       ページアップ・ページダウン時に        *
*       マウスの座標から新しい値を計算する    *
*******************************************/
NewScValue( selHdl, pt )
control **selHdl;
point_t  pt;
{
        rect r;
        int  len,pos,max,v;

        r=winPtr->wGraph.grRect;
        max=CMMaxGet( selHdl );

        if( selHdl==scBarV ){
            len=r.bottom-r.top-SCBAR_WIDTH;
            pos=pt.p.y-r.top;
        }
        else{
            len=r.right-r.left-SCBAR_WIDTH;
            pos=pt.p.x-r.left;
        }
        v=(pos*max)/len;
        return( max-v );
}
/*******************************************
*       ウィンドウのサイズが変化したとき      *
*       スクロールバーの最大値を変更する      *
*******************************************/
ChangeMaxValue()
{
            rect    r;
        int w,h;
        int vv,vh;

            r=winPtr->wGraph.grRect;
        w=r.right  - r.left - SCBAR_WIDTH;
        h=r.bottom - r.top  - SCBAR_WIDTH;

        vv=CMValueGet( scBarV );
        vh=CMValueGet( scBarH );

        if((vv+h)<=DEST_HIGHT)
            CMMaxSet( scBarV, DEST_HIGHT-h );
        else
            CMMaxSet( scBarV, maxLimV );

        if((vh+w)<=DEST_WIDTH)
            CMMaxSet( scBarH, DEST_WIDTH-w );
        else
            CMMaxSet( scBarH, maxLimH );

            UpdScBar(1);
}
/*************************************
*       スクロールバーの処理をする     *
*************************************/
procScBar( part, selHdl, pt )
int     part;
control **selHdl;
point_t pt;
{
        point_t ptu,ptd;
        int     v,d;
        int     mx,new;

        if( selHdl==scBarV ){
            ptu.x_y=POINT(0,-1);    /* 垂直バー アップ時の方向 */
            ptd.x_y=POINT(0,1);     /* 垂直バー ダウン時の方向 */
        }
        else{
            ptu.x_y=POINT(-1,0);    /* 水平バー アップ時の方向 */
            ptd.x_y=POINT(1,0);     /* 水平バー ダウン時の方向 */
        }
        v =CMValueGet( selHdl );     /* スクロールバーの値を取り出す */
        mx=CMMaxGet( selHdl );       /* スクロールバーの最大値を取り出す */

        switch(part){
        case C_INDOWN:
                    d=24;
                    if((v+d)>mx) d=mx-v;
                    CMValueSet( selHdl, v+d );
                    ptd.p.x*=d;
                    ptd.p.y*=d;
                    DoScroll(ptd);
                    drawGrowBox();
                    break;
        case C_INUP:
                    d=24;
                    if((v-d)<0) d=v;
                    CMValueSet( selHdl, v-d );
                    ptu.p.x*=d;
                    ptu.p.y*=d;
                    DoScroll(ptu);
                    drawGrowBox();
                    break;
        case C_INPGUP:
        case C_INPGDOWN:
                    new=NewScValue( selHdl, pt );
                    CMValueSet( selHdl, new );
                    ptd.p.x=ptd.p.x*(new-v);    /* スクロールは値の差
分だけ */
                    ptd.p.y=ptd.p.y*(new-v);
                    DoScroll(ptd);
                    drawGrowBox();
                    break;
        case C_INTHUMB:
                    CMCheck( selHdl, pt, (void(*))0);
                    new=CMValueGet( selHdl );
                    ptd.p.x=ptd.p.x*(new-v);
                    ptd.p.y=ptd.p.y*(new-v);
                    DoScroll(ptd);
                    drawGrowBox();
                    break;
            }
}

main()
{
        if( SX_init()==FALSE ){
            DMError(0x101,"ウィンドウかオープンできません");
            exit();
        }
        while( 1 ){
            TSEventAvail(EVENTMASK,(tsevent*)&eventRec);
            switch( eventRec.eWhat ){
            case E_IDLE:        procIDLE();     break;
            case E_MSLDOWN:     procMSLDOWN();  break;
            case E_MSLUP:       procMSLUP();    break;
            case E_MSRDOWN:     procMSRDOWN();  break;
            case E_MSRUP:       procMSRUP();    break;
            case E_KEYDOWN:     procKEYDOWN();  break;
            case E_KEYUP:       procKEYUP();    break;
            case E_UPDATE:      procUPDATE();   break;
            case E_ACTIVATE:    procACTIVATE();break;
            case E_SYSTEM1:     procSYSTEM();   break;
            case E_SYSTEM2:     procSYSTEM();   break;
            case E_USER1:       procUSER();     break;
            case E_USER2:       procUSER();     break;
            }
        }
}

SX_init()
{
        task    taskBuf;

        TSGetTdb(&taskBuf, -1);
        if( (TSTakeParam(&taskBuf.command,&winSize,NULL,0,NULL,N
ULL)&1)==0 ){
            *(int *)&winSize.left = TSGetWindowPos();
            winSize.right  = winSize.left + WINWIDTH;
            winSize.bottom = winSize.top  + WINHIGHT;
        }
        winPtr=WMOpen(NULL,&winSize,(LASCII*)WINTITLE,TRUE,WINDE
FID,(window *)-1,TRUE,TSGetID());
        if( winPtr == NULL ) return( FALSE );
        winPtr->wOption = WINOPT;
        activeFlag=FALSE;
        ctrlFlag  = CtrlPrepare();/* コントロールが不要なら ctrlFlag=FA
LSE */
        DRAW();
        drawGrowBox();
        return( TRUE );
}
```

▶最近，打ち込もうと思っていたプログラムの数々を打ち込み始めました。心なしか私のコンピュータが元気になったように見えます。今日も頑張って入力しようっと。

伊藤　徹(22)千葉県

```
SX_term()
{
    if( ctrlFlag ) CtrlDispose();
    WMDispose( winPtr );
    exit();
}

drawGrowBox()
{
    GMSetGraph( &winPtr->wGraph );
    WMDrawGBox( winPtr );
}

CtrlPrepare()
{
    MkScBar();
    return( TRUE );
}

CtrlDispose()
{
    CMDispose( scBarV );
    CMDispose( scBarH );
    return( TRUE );
}

MenuPrepare()
{
    return( FALSE );
}

MenuDispose()
{
    return( FALSE );
}

procIDLE()
{
    return( FALSE );
}

procMSLDOWN()
{
    point_t pt;
    control **selHdl;
    rect    destRect;
    int     part;

    if( (window*)eventRec.eWhom != winPtr ) return( FALSE );
    if( activeFlag == FALSE ){
        WMSelect( winPtr );
        activeFlag = TRUE;
        if( EMLStill() == 0) goto checkDClick;
    }
    pt.p.x=winPtr->wGraph.grRect.left;/* ズームイン・ズームアウト時の
 */
    pt.p.y=winPtr->wGraph.grRect.top; /* ホーム位置の補正用 */
    switch( SXCallWindM(winPtr,(tsevent*)&eventRec) ){
    case W_INCLOSE:
        SX_term(); break;
    case W_INGROW:
    case W_INZMOUT:
    case W_INZMIN:
        GMSetHome( pt );
        ChangeMaxValue();
        GMClipRect( &winPtr->wGraph.grRect );
        break;
    case W_ININSIDE:
#ifdef  CTRL_MAN
        destRect.left  =0;
        destRect.top   =0;
        destRect.right =DEST_WIDTH;
        destRect.bottom=DEST_HIGHT;
        part=SXCallCtrlM(winPtr,(tsevent*)&eventRec,
             (void**)scBarV,(void**)scBarH,&destRect,&selHdl
);
#else
        pt.x_y=GMGlobalToLocal( eventRec.eWhere );
        part=CMFind( pt, winPtr, &selHdl );
        if( selHdl==scBarV || selHdl==scBarH )
            procScBar( part, selHdl, pt);  /* スクロールバーの処理
 */
#endif
        break;
    }

checkDClick:
    TSGetEvent(EVENTMASK,(tsevent*)&eventRec);
    return( TRUE );
}

procMSLUP()
{
    return( FALSE );
}

procMSRDOWN()
{
#ifdef     MENU
    menu  **menuHdl;
    int    item;

    if( ((window*)(eventRec.eWhom)!=winPtr)
       ||(activeFlag==FALSE) )
        return( FALSE );
    GMSetGraph( &winPtr->wGraph );
    menuHdl=MNConvert(0,"ものろおぐ",0);
    item=MNSelect( menuHdl, eventRec.eWhere );
    TSGetEvent( EVENTMASK, (tsevent*)&eventRec );
    if(item==1) doDialog();
    MMHdlDispose( menuHdl );
    return( TRUE );
#else
    return( FALSE );
#endif
}

procMSRUP()
{
    return( FALSE );
}

procKEYDOWN()
{
    return( FALSE );
}

procKEYUP()
{
    return( FALSE );
}

procUPDATE()
{
    if( (window*)eventRec.eWhom != winPtr ) return( FALSE );
    WMUpdate( winPtr );
    DRAW();
    if( ctrlFlag ) CMDraw( winPtr );
    WMUpdtOver( winPtr );
    drawGrowBox();
    TSGetEvent(EVENTMASK,(tsevent*)&eventRec);
}

procACTIVATE()
{
    if( (window*)eventRec.eWhom == winPtr )    activeFlag =
TRUE;
    else if( eventRec.eWhom != NULL ){
        if( activeFlag ) {
            activeFlag = FALSE;
            TSGetEvent(EVENTMASK,(tsevent*)&eventRec);
        }
    }
    return( TRUE );
}

procSYSTEM()
{
    switch( ((tsevent*)&eventRec)->what2 ){
    case CLOSEALL:
    case ENDTSK:
        SX_term(); break;
    case WINDOWSELECT:
        WMSelect( winPtr ); break;
    }
}

procUSER()
{
    return( FALSE );
}

DRAW()
{
#define SIZE    24

    point_t pt;
    point_t size;
    int kind;
    rect    clip;

    GMSetGraph( &winPtr->wGraph );

    clip=winPtr->wGraph.grRect;
    clip.right-=18;
    clip.bottom-=18;
    GMClipRect( &clip );

    pt.x_y=(SIZE<<16)|SIZE;
    size.x_y=GMPenSize( pt );
    kind=GMFontKind( G_ROM24 );

    pt.x_y=0*SIZE;
    GMMove( pt );
    GMDrawStrZ("あいうえおかきくけこさしすせそたちつてと");

    pt.p.y=1*SIZE;
    GMMove( pt );
    GMDrawStrZ("なにぬねのはひふへほまみむめもやゆよわをん");

    pt.p.y=2*SIZE;
    GMMove( pt );
    GMDrawStrZ("ABCDEFGHJIKLMNOPQRTSU");

    pt.p.y=3*SIZE;
    GMMove( pt );
    GMDrawStrZ("VWXYZabcdefghijklmnop");


    pt.p.y=4*SIZE;
    GMMove( pt );
    GMDrawStrZ("qrstuvwxyz÷=≠<>≦≧∞∴♂♀");

    pt.p.y=5*SIZE;
    GMMove( pt );
    GMDrawStrZ("°′″℃¥$¢£%#&*@§☆★○●◎◇◆");

    pt.p.y=6*SIZE;
    GMMove( pt );
    GMDrawStrZ("□■△▲▽▼※〒→←↑↓〓∈∋⊆⊇⊂⊃∪∩");
```

▶景気の衰退，株価の低迷……国の対策もあてにはならない。しかし，私は阪神タイガースが優勝するとすべて解決しそうな気がします。えっ，そんなことを思うのは私だけですか？
坊農　誠(21)福井県

```
        pt.p.y=7*SIZE;
        GMMove( pt );
        GMDrawStrZ("∧∨¬⇒⇔∀∃∠⊥⌒∂∇≡≒≪≫√∽∝∵∫");

        pt.p.y=8*SIZE;
        GMMove( pt );
        GMDrawStrZ("∬Å‰♯♭♪†‡¶◯、。，．・：；？！゛゜");

        pt.p.y=9*SIZE;
        GMMove( pt );
        GMDrawStrZ("´｀¨＾￣＿ヽヾゝゞ〃仝々〆〇ー―‐／＼～∥");

        pt.p.y=10*SIZE;
        GMMove( pt );
        GMDrawStrZ("がぎぐげござじずぜぞだぢづでどばびぶべぼぱ");

        pt.p.y=11*SIZE;
        GMMove( pt );
        GMDrawStrZ("ぴぷぺぽゐゑヰヱヲヴΘΞΠΣΥΦΨΩαβγ");

        pt.p.y=12*SIZE;
        GMMove( pt );
        GMDrawStrZ("δεζηθικλμνξοπρστυφχψω");

        GMPenSize( size );
        GMFontKind( kind );
        GMClipRect( &winPtr->wGraph.grRect );
}
```

リスト2

```
===============  SX2  ==================
/*
        スクロールバーの値に応じて
        描画する側が描く位置をずらす
        プログラムの例

                                1992.8.9 中森 章
*/
#include <stdio.h>
#define __POINT_T
#include <sxlib.h>

#define DLOG_WIDTH  400
#define DLOG_HIGHT  200
#define SCBAR_MIN   0
#define SCBAR_MAX   176

doDialog()
{
        rect    dispRect;
        rect    winRect;
        rect    btnRect;
        rect    scbRect;

        point_t pt;
        window  *dlogPtr;
        control **scbar;
        control **okbtn;
        control **selHdl;
        int part;
        int value;

        event   eventRec;

        region  **updRgn;
        rect    scrRect;


        SXGetDispRect( &dispRect );
        pt.x_y=(DLOG_WIDTH<<16)|DLOG_HIGHT;
        GMCenterRect( &winRect, &dispRect, pt, 0);

        dlogPtr=WMOpen(NULL,&winRect,(LASCII*)"¥012ダイアログ",-1,(
0x27<<4),
                        (window *)-1,-1,TSGetID());
        GMSetGraph( &dlogPtr->wGraph );

        scbRect.left   = DLOG_WIDTH-36;
        scbRect.top    = 18;
        scbRect.right  = DLOG_WIDTH-18;
        scbRect.bottom = DLOG_HIGHT-40;

        value=SCBAR_MIN;
        scbar=CMOpen(dlogPtr,&scbRect,(LASCII*)"",-1,
                    value,
                    SCBAR_MIN,SCBAR_MAX,CI_SCRLBRV<<4,0);

        btnRect.left   = DLOG_WIDTH-60;
        btnRect.top    = DLOG_HIGHT-30;
        btnRect.right  = DLOG_WIDTH-18;
        btnRect.bottom = DLOG_HIGHT-12;
        okbtn=CMOpen(dlogPtr,&btnRect,(LASCII*)"¥007 確 認 ",-1,0
,
                    0,1,CI_STDBTN<<4,0);
        CMDraw(dlogPtr);

        scrRect.left   = 15;
        scrRect.top    = 15;
        scrRect.right  = DLOG_WIDTH-40;
        scrRect.bottom = DLOG_HIGHT-40;

        GMShadowRect( &scrRect );

        scrRect.left   += 3;
        scrRect.top    += 3;
        scrRect.right  -= 3;
        scrRect.bottom -= 3;

        GMClipRect( &scrRect );
        dlogDraw(value);
        GMClipRect( &dlogPtr->wGraph.grRect );

        while(1){
            EMGet(EM_EVERY,&eventRec);
            switch( eventRec.eWhat ){
            case E_MSLDOWN:
                pt=eventRec.eWhere; /* マウスの座標 */
                pt.x_y=GMGlobalToLocal(pt);
                part=CMFind(pt,dlogPtr,&selHdl);
                if(selHdl==okbtn){
                    CMCheck(okbtn,pt,(void*)0);
                    goto endLoop;
                }
                else if(selHdl==scbar){
                    value=CMValueGet(scbar);
                    pt.x_y=0;
                    switch(part){
                    case C_INDOWN:
                        if((value+12)<=SCBAR_MAX){
                            value+=16;
                            pt.p.x=0;
                            pt.p.y=-16;
                        }
                        break;
                    case C_INUP:
                        if((value-12)>=SCBAR_MIN){
                            value-=16;
                            pt.p.x=0;
                            pt.p.y=16;
                        }
                        break;
                    }
                    CMValueSet(scbar,value);
                    updRgn=GMNewRgn();
                    GMScroll( &scrRect, pt, updRgn );
                    pt.x_y=0;
                    pt.x_y=GMLocalToGlobal( pt );
                    GMSlideRgn( updRgn, pt );/* リージョンをグローバル座
標系に変換 */
                    dlogPtr->wUpdate=updRgn; /* アップデートリージョンを
直接指定 */
                    WMUpdate( dlogPtr );
                    dlogDraw(value);
                    WMUpdtOver( dlogPtr );
                    GMDisposeRgn( updRgn );
                }
                break;
            case E_KEYDOWN:
                if((short)(eventRec.eWhom)!=13)
                    break;
                CMShine(okbtn,C_INBTTN);
                goto endLoop;
            }
        }
    endLoop:
        CMDispose(okbtn);
        CMDispose(scbar);
        WMDispose(dlogPtr);
}

dlogDraw(value)
int value;
{
        int ix;
        point_t pt,size;
        int kind;
        char    *msg[]={
            "わたし月野うさぎ。14歳、中2、蟹座のO型。",
            "誕生石は真珠。",
            "性格は、人よりちょおっとおっちょこちょいで",
            "ちょっと泣き虫ってとこかな。",
            "あるとき、へんてこりんな黒猫ルナが現れて",
            "セーラー服戦士にしてくれたんだけど、",
            "悪いやつらと戦えなんて",
            "不安たらたらって感じ。",
            "でもまっ。なんとかなるか。",
            "",
            "わたし月野うさぎ。こう見えても正義の味方。",
            "不思議なブローチで変身して悪い奴らと戦うの。",
            "ごぞんじ、愛と正義のセーラー服美少女戦士",
            "セーラームーン。そして、わたしのお友達は、",
            "心優しき天才少女水野亜美ちゃんと、",
            "ちょっぴりおこりんぼの",
            "霊感少女火野レイちゃん。",
            "ふたりともしっかりしてて負けそう。",
            "でもまっ、やるっきゃないない。",
            (char*)NULL
        };
        pt.x_y=(16<<16)|16;
        size.x_y=GMPenSize( pt );
        kind=GMFontKind( G_ROM16 );
        pt.p.x=18;
        pt.p.y=18;
        ix=value/16;
        while(msg[ix]){
            GMMove( pt );
            GMDrawStrZ( msg[ix] );
            pt.p.y+=16;
            ix++;
        }
        GMPenSize( size );
        GMFontKind( kind );
}
```

THE SENTINEL

〈対応機種一覧〉●MZ-80K/C/700/1500●MZ-80B/2000●MZ-2500/2861●X1●X1turbo/Z●PC-8001/8801/88●SMC-777/C●PASOPIA/5●PASOPIA 7●FM-7/77/AV●PC-286/386/9801/98●X68000
掲載されたプログラムの利用には各機種用のS-OS"SWORD"システムが必要です。

第125部　SLENDER HUL　実践Small-C講座(6)

●SLENDER HUL

初のSmall-C本格アプリケーションをようやくお届けします。内容は横スクロールタイプのシューティングゲームで，作者はお馴染みの柴田氏です。

作りは結構オーソドックスなもので，柴田氏お得意のキャラグラアニメーションが光る作品。そして，このゲームの最大の特徴は，シールドエネルギーが兵器エネルギーを兼ねている点です。

ビームを連続で撃てばどんどんシールドエネルギーが減っていき，ダメージを受けたときにシールドエネルギーがなくなるとゲームオーバー。

これによって，シューティングゲームの撃ちっぱなしで敵を蹴散らす爽快感ではなく，正確な狙い撃ちが要求されるパズル的な要素が重要視されているのです。

無駄を省いた正確な操作で，効率よく敵を撃ち落としていく。画面はシンプルながら結構熱くなれるゲームに仕上がっています。かなり，リストの分量はあります。がんばって打ち込んで遊んでみましょう。

●Small-Cによるアプリケーション

さて，いままでも，石上氏がちょっとしたプログラムを紹介してきましたが，あくまでサンプルプログラムの域を出ていないものばかり。ちょっともの足りない思いで，Small-Cを使っている人も多かったことでしょう。もしかして，Small-Cってそんなに……などと思ってしまっている人もいるかもしれません。

確かにSmall-Cのみで，すべてをまかなおうとすると無理がある場合もあります。そんなときには，「SLENDER HUL」のように処理の重たい部分をインラインアセンブラで，記述してしまえばいいのです。

引数の受け渡し方法で，ちょっとしたルールがありますが，それさえ押さえておけば簡単に必要な部分をアセンブラに置き換えることができます。中核となるメインルーチンはC言語で，高速化するところのみアセンブラを使う。C言語らしい使い方ですね。

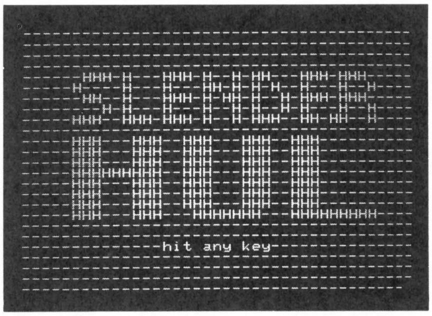
いままで本格的なプログラムがなかったので，これからSmall-Cを本格的に使おうとしている人にとって，「SLENDER HUL」には参考となる部分が多いでしょう。そして，盗んだテクニックを使って何か作ってみませんか。

●S-OSの系譜(37)

1988年12月号では，Z80ソースジェネレータ「SOURCERY」が発表されました。

ソースジェネレータとは，マシン語コードからアセンブラコードを生成するもので，1985年8月号の「ZING」以来，3年ぶりの登場となったのです。

この「SOURCERY」は，「ZING」の機能強化版といった感じで，特にプログラム中のデータエリアが機能別に管理することができ，かなり忠実なソースリストを再現できたのです。



もちろん，データエリアの自動判別などという高級なことはできませんでした。すべて手作業でワークを見つけるなりしなければなりませんが，それでも便利な機能でした。

プログラムの解析，改造をしない人にとって，ソースが公開されているS-OSではあまりありがたみのないプログラムではないか？　と早合点するかもしれません。要するに，ダンプリストではなくソースリストを打ち込めばすんでしまうからです。

しかし，入力する速度，チェックを考えると圧倒的にCRCチェックサム付きのダンプリストに軍配が上がります。無謀にも，数千行におよぶソースリストを打ち込もうとして挫折してしまった，そんな苦い思い出を持っている人もいることでしょう。

また，"ないものは自分で作る"精神を持ったユーザーにとって，アプリケーションのカスタマイズをしようとしたとき，ソースジェネレータは非常に便利なものでした。ソースリストがあれば，わざわざパッチ当てのプログラムを考慮することなく，簡単に思いのままの機能を付加することができたのです。

1992■インデックス

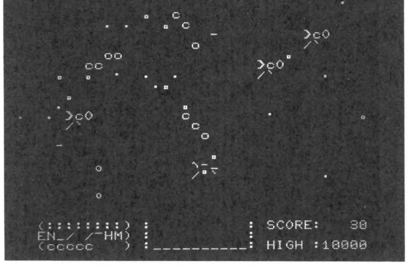
■92年1月号
第115部　LINER
■92年2月号
第116部　シミュレーションゲームPOLANYI
■92年3月号
第117部　カードゲームKLONDIKE
■92年4月号
第118部　オプティマイザO80実践Small-C講座(1)
■92年5月号
第119部　COMMAND.OBJ実践Small-C講座(2)
■92年6月号
第120部　COMMAND.OBJ2実践Small-C講座(3)
■92年7月号
第121部　関数リファレンス実践Small-C講座(4)
■92年8月号
第122部　ワイルドカード実践Small-C講座(5)
第123部　グラフィックライブラリ GRAPH.LIB
■92年9月号
第124部　O-EDIT&MODCNV

全機種共通
S-OS"SWORD"要

SLENDER HUL
実践Small-C講座(6)

Shibata Atsushi
柴田　淳

Small-Cを使った横スクロールタイプのシューティングゲームの登場です。Small-C発表以来，初めての本格的なアプリケーション。ボリュームたっぷりですが，気合一発入力しましょう。



プログラミングをするというのは，デバッグをするということと99%同義語です。デバッグをしないままのプログロラムというのは，たいていバグの固まりであり，自分で使うのならともかく，人に見せるものとしてはふさわしくありません。

そこで，デバッグとはそもそもどういうことなのかと考えてみると，要するに，人間が機械の上をいくということだと思うのです。特にマシン語レベルのプログラムを組もうとした場合などは，CPUの動きをなるべく詳しく知っておかなければなりません。そうしないと，重大な局面に直面したときに動きがとれなくなってしまいます。そしてデバッグが不可能になった瞬間，それまで積み重ねてきた苦労は水泡に帰するのです。どんなに小さなバグであっても。

実をいうと，僕はこのプログラムを組むまで，Cについてはほとんど何も知りませんでした。だからデバッグをしようとしても，コンパイラが吐き出すコードの動きが読めず，バグが出るたびに頭がこんがらがって結構ひどい目にあったりしました。

繰り返しになるけど，プログラミングというのは，デバッグの積み重ねである，と僕は思います。バグを見つけるたびに，何が原因かを根気強く，ひとつひとつ論理的な証拠を積み重ねていって導き出し，消して，書いて，また消して，新しいバグができたりして，その過程は決して楽しいものではないけれど，すべてのバグを取り除いたとき，えもいわれぬ充実感に包まれるのです。

で，何も知らないでCに飛び込んだ僕の場合，プログラミングが終わったとき，ふと我に返ると，結構Cに詳しくなった自分を見つけてなんだか得した気分になったものでした。うーん，Cにはこういう勉強法もあったのだなあ。とても人にはオススメはできないけれど。

入力方法とゲームの遊び方

入力方法は，リスト1～28を入力して，WLKでひとつにまとめればいいだけです。これだけではあまりにも不親切なので，もう少し詳しく書いておきましょう。

まず，このプログラムを入力する前に，1991年の12月号に載った「Small-C SLANGコンパチ関数」が必要です。まだ入力していない人はこの機会に入力しましょう。なお，プログラミング中にバグをひとつ発見しました。カコミを見てデバッグしておいてください。

ライブラリを用意したらエディタなどからリストを入力して，いったんディスクにセーブします。終わったら「.C」拡張子の付いたリストは，コンパイル後WLBでアセンブル，「.ASM」拡張子の付いたリストは，WZDでアセンブルのみします。1枚のディスクで作業していると容量がいっぱいになってしまうので，何枚かのディスクに分けて作業したほうがいいでしょう。

すべてのリストにつき，コンパイル，アセンブルを終了させたら，出来上がった「.REL」拡張子の付いたファイルをすべてひとつのディスクに集め，さらにWLKとclib.LIBも同じディスクにセーブします。そこまでの作業が終わったら，コマンドラインから，

```
# WLK
*/P:3000, MAIN, enemy, enbl1
*enbl2, plsh, indc, stit, smass
*chhx, jp, ind, title, enorg
*sp, scr, pr, msg, axy, dx, dy
*strs, plbul, blnk, move, cls2
*smoke, PRN, ADD, slnlib/S, clib/S
*MAIN/N:P
```

と入力すれば，作業は終了です。ファイルを読み込み，スタートアドレスにジャンプすればゲームが始まります。

さて遊び方ですが，この「SLENDER HUL」は基本的にシューティングゲームです。テンキーで自機を動かし，Zキーで弾を打ちます。また，Xキーを押すことによって，ホーミングミサイルをストックできます。ストックがあるときにスペースキーを押せば，近場の敵を目がけてハデハデのミサイルが飛んでいきます。ゲーム中にはシールドエネルギーが一定の度合いで増えていき，このシールドは敵に接触したり弾を受けたりすると減るようになっており，すべてなくなるとゲームオーバーです。

そして，ここからがこのゲームの独創的なところ（と自分では思っている）です。いま述べたシールドエネルギーというのは，実はダメージを受けたときにだけ減るのでなく，自機の弾を撃ったり，ミサイルをス

トックしたりすることによっても減ってしまいます。つまり，経済的な攻撃をしないとシールドがすぐ底をついてしまうわけです。いうなれば，このゲームは，シューティングゲーム史上初めて，ゲームシステムに経済原則を導入した画期的な作品なのです（本当かなあ）。

プログラムについて

今回のゲームでは，「Small-C」というお題目があったにもかかわらず，かなりの部分をアセンブラで書いています。全部Cで書けなくもなかったのだけど，速度の問題もあったし，単純な作業はマシン語で書いたほうがいろいろ利点もあるのです。

ここでいう単純な作業というのは，主に処理のなかで掛け算とか割り算とかを使わないですむもののことです。星を左に流したり，仮想画面を書き換えたりするのがそれに該当します。わかる人はわかると思いますが，こういう処理はアセンブラで書いたほうが気楽なのです。ただし，Cから呼び出す関数を書くのですから，ある程度の決まりごとは守らなければなりません。

＊　　＊　　＊

プログラミングは旅に似ています。効率よく作業を進めるためには，あらかじめ確かな目標点を設定し，まるで旅の前に時刻表を見るのと同じように，その点にたどり着くための綿密な計画を立て，それから初めて作業に取り掛からねばなりません。

「まず部分から作っていく」

Cのプログラミングをひと言で表すとこうなるでしょうか。で，すべての「部分」が完成したら，えいやっ，とばかりにそれらを統合し，プログラムは初めて完成をみるのです。

計画どおりに目的地にたどり着いたときの達成感というか満足感というか，プログラミングはデバッグであり，デバッグは根気であるわけです。で，この種の満足は，根気を持続させるためにとても大きな役割を果たしていると，僕は思います。

プログラミングは旅に似ています。だから，ひとつのプログラムを完成させると，思い出がひとつ，出来上がるのです（とクサイことをいって去る）。

Small-C SLANG コンパチ関数のバグ

関数rnd_sub()にバグがあるようです。11行の，

LD　HL,(HL_BUFF)

を

LD　HL, HL_BUFF

に変更してください。要するにカッコを取るだけです。

リスト1

```
/* MAIN.C */

/*------------------------------

  Shooting Game  for  Small-C

    1992  Programmed by Ats.

-------------------------------*/

main()
{
  int  high,sco,tm2;
  char key,x,y;
  char scs[15],enm[256];
  char ebl[160],eb2[160];
  char pl[40];
  char pbl[220],phm[120];
  char str[120],dst[640];
  stit( str );
  ind( enm , 256 );
  ind( ebl , 160 );
  ind( eb2 , 160 );
  ind( dst , 640 );
  ind( pbl , 220 );
  ind( phm , 120 );
  ind( pl  , 40  );
  high = 1000;
  widch( 40 );

 while( high != 0 )
 {
  open();

  pl[1] = 20; pl[2] = 10;
  pl[9] =  1; pl[10] = 0;
  pl[5] = 16;
  enorg ( 100 );
  cls();
  scit( high );
  scr();
  key = 0; sco = 0;

  while ( pl[9] != 0 )
   {
     cls2( 0 );
     strs( str );
     x =( key == 'S' );
     key = inkey( x );
     pler( pl,pbl,phm,dst,enm,key );
     indc( pl );
     plbul( pbl );
     plhm( phm,enm,dst );
     bmpl( pbl,phm,enm,dst,pl );
     enemy( enm,ebl,eb2,pl );
     enbl1( ebl );
     enbl2( eb2,pl );
     bmen( ebl,eb2,enm,pl,dst );
     smoke( dst );
     tm2 = enorg( enm );
     if ( tm2 == -1 )
     {
        enorg( 50 );
        pl[0]++;
     }
     sco += pl[38];
     itod( sco,scs,5 );
     strcat( scs,"0" );
     msg( 32,22,scs );
     scr();
   }
     x = pl[1]; y = pl[2];
     smass( dst,x+1,y  ,8,12,1,8 );
     smass( dst,x  ,y+1,8,12,1,8 );
     smass( dst,x+1,y+2,8,12,1,8 );
     smass( dst,x+2,y  ,8,16,1,8 );
     smass( dst,x+1,y+1,8,16,1,8 );
     smass( dst,x+1,y+2,8,12,1,8 );
     strcpy( scs,"GAME OVER" );
   for ( tm2 = 0; tm2 < 100; tm2++ )
   {
     cls2( 0 );
     strs( str );
     enemy( enm,ebl,eb2,pl );
     enbl1( ebl );
     enbl2( eb2,pl );
     smoke( dst );
     msg( 14,10,scs );
     scr();
   }
   if ( high < sco )
     high = sco;
  }
}


pler( pl,pbl,phm,dst,enm,key )
 char *pl,*pbl,*phm,*dst,*enm,key;
 {
  char a,b,c;

     move( pl );
     if ( pl[1] == -1 )
       pl[1] = 0;
     if ( pl[1] == 38 )
       pl[1] = 37;
     if ( pl[2] == -1 )
       pl[2] = 0;
     if ( pl[2] == 21 )
       pl[2] = 20;
     a = 0; b = 0; c = 0;
     if ( key == '7' || key == '8'
       || key == '9' ) {
       b = -4; c = 1;  }
     if ( key == '1' || key == '2'
       || key == '3' ) {
       b =  4; c = 2;  }
     if ( key == '1' || key == '4'
       || key == '7' )
       a = -4;
     if ( key == '3' || key == '6'
       || key == '9' )
       a =  4;
     if ( a != 0 )
       pl[3] = chhx( pl[3],a );
     else
       pl[3] /= 2;
     if ( b != 0 )
       pl[4] = chhx( pl[4],b );
     else
```

```c
            pl[4] /= 2;
        sp ( pl[1],pl[2],c );
        pl[9]++;
        if ( pl[9] > 63 )
            pl[9] = 63;
        if ( pl[9] < 0 )
            pl[9] = 0;
        if ( key == 'Z' && pl[9] > 4+pl[0] )
        {
            plsh( pl,pbl,dst );
            pl[9] -= 2+pl[0];
        }
        if ( key=='X' && pl[9]>5+pl[0]
                      && pl[10]<8 )
        {
            pl[11] += 3; pl[9] -= 4+pl[0];
            if ( pl[11] > 8 )
            {
                pl[11] = 0;
                pl[10]++;
                jp( 9-pl[10],7,pl );
            }
        }
        if ( key == ' ' )
            pl[12] = 1;
        if ( pl[12] > 0 && pl[10] > 0 )
        {
            pl[12]++;
            if ( pl[12] % 5 == 2 )
                hmsh( pl,phm,enm );
        }
        else
            pl[12] = 0;
}

hmsh( pl,phm,enm )
char *pl,*phm,*enm;
{
 char  x,y,a,b;
 int   i,j,c,d;
    c = 3000; j = -1;
    x = pl[1]; y = pl[2];
    for ( i = 0; i < 256; i += 16 )
    {
     if ( enm[i] != 0 && enm[i+13] == 0 )
     {
        a = x - enm[i+1]; b = y - enm[i+2];
        d = a * a + b * b;
        if ( c > d )
        {
            j = i; c = d;
        }
     }
    }
    if ( j != -1 )
    {
     c = blnk( phm,10,12 );
     if ( c != -1 )
     {
      enm[j+13] = 1;
      phm[c  ] = 1; phm[c+1] = x;
      phm[c+2] = y; phm[c+5] = 0;
      phm[c+9] = 8;
      phm[c+10] = j / 16;
      pl[10]--;
      jp( 1+((pl[12]/5) % 1)*2,8,pl );
     }
    }
    else
     pl[12] = 0;
}

plhm( phm,enm,dst )
char *phm,*enm,*dst;
{
 char a,b,c;
 int  i,j;
    for ( i = 0; i < 120; i += 12 )
    {
     if ( phm[i] != 0 )
     {
        a = dx(phm[i+9]); b = dy(phm[i+9]);
        phm[i+3] = a; phm[i+4] = b;
        move( &phm[i] );
        move( &phm[i] );
        if ( phm[i+10] == 17 )
        {
         if ( phm[i+1]>39 || phm[i+1]<0
           || phm[i+2]>21 || phm[i+2]<0 )
          phm[i] = 0;
        }
        else
        {
         j = phm[i+10] * 16;
         if ( enm[j] == 0 )
         {
          phm[i+10] = 17; phm[i+5]  = 16;
         }
         if ( phm[i+5] != 16 )
          phm[i+5] += 2;
         else
         {
          a = enm[j+1]; b = enm[j+2];
          c = dr( phm[i+1],phm[i+2],a,b );
          phm[i+9] = apr( phm[i+9],c );
         }
        }
        a = phm[i+1]; b = phm[i+2];
        c =(phm[i+9] + 8 ) % 16;
        smass( dst,a,b,c,16,32,16 );
        smass( dst,a,b,c, 8,32,16 );
        if ( b < 21 )
         pr( a,b,'o' );
     }
    }
}

bmpl( pbl,phm,enm,dst,pl )
char *pbl,*phm,*enm,*dst,*pl;
{
 int i,j;
 char a,b,x,y,xy[2];
     pl[38] = 0;
     for ( i = 0; i < 220; i += 10 )
     {
      if ( pbl[i] != 0 )
      {
       x = pbl[i+1]; y = pbl[i+2];
       for ( j = 0; j < 256; j += 16 )
       {
        if ( enm[j] != 0 )
        {
         a = enm[j+1]; b = enm[j+2];
         axy( xy,enm[j] );
         if ( x >= a && x-a < xy[0]+1
           && y >= b && y-b < xy[1] )
         {
           pbl[i] = 0;
           enm[j+12]--; pl[38] += 1;
           if ( enm[j+12] < 1 )
            bomb( enm,dst,pl,j,0 );
           else
           {
            smass( dst,x,y,0,16,32,16 );
            jp( 2,3,pl ); jp( 7,3,pl );
           }
           break;
         }
        }
       }
      }
    }
    for ( i = 0; i < 120; i += 12 )
    {
     if ( phm[i] != 0 )
     {
      j = phm[i+10] * 16;
      a = enm[j+1]; b = enm[j+2];
      axy( xy,enm[j] );
      x = phm[i+1]; y = phm[i+2];
      if ( x >= a && x-a < xy[0]+1
        && y >= b && y-b < xy[1] )
      {
        phm[i] = 0;
        enm[j+12] -= 8;
        enm[j+13] = 0; pl[38] += 8;
        if ( enm[j+12] < 1 )
         bomb( enm,dst,pl,j,phm[i+9] );
        else
        {
         smass( dst,x,y,phm[i+9],16,32,16 );
         jp( 1,5,pl ); jp( 8,5,pl );
        }
      }
     }
    }
}

bmen( eb1,eb2,enm,pl,dst )
char *eb1,*eb2,*enm,*pl,*dst;
{
    int i,j;
    char a,b,x,y,xy[2];
    x = pl[1]; y = pl[2];
    for ( i = 0; i < 160; i += 10 )
    {
        if ( eb1[i] != 0 )
        {
         a = eb1[i+1];
         if ( eb1[i+2] == y
          && a >= x && a - x < 3 )
         {
     pl[3] = chhx(pl[3],8*sign(eb1[i+3]));
           pl[9] -= 10;
           eb1[i] = 0;
           jp( 3,5,pl ); jp( 6,6,pl );
         }
        }
        if ( eb2[i] != 0 )
        {
         a = eb2[i+1];
         if ( eb2[i+2] == y
           && a >= x && a - x < 3 )
         {
     pl[4] = chhx(pl[4],8*sign(eb2[i+4]));
          pl[9] -= 18;
          eb2[i] = 0;
          jp( 2,5,pl ); jp( 7,4,pl );
         }
        }
    }
    for ( i = 0; i < 256; i += 16 )
    {
     if ( enm[i] != 0 )
     {
       axy( xy,enm[i] );
       a = enm[i+1]-1; b = enm[i+2]-1;
       if ( x >= a && x-a < xy[0]+1
         && y >= b && y-b < xy[1]+1 )
       {
        j = enm[i+12]*5;
         if ( j > 48 )
          j = 48;
        pl[9] -= j;
        bomb( enm,dst,pl,i,0 );
       }
     }
    }
}
```

▶数値演算プロセッサを買おうかなあ，と思っていたところに９月号の特集。う～んタイムリーだなあ。ますます数値演算プロセッサが欲しくなりました。早くお金を貯めなきゃ。

伊藤　直樹(24)愛知県

```
bomb( enm,dst,pl,j,b )
char *enm,*dst,*pl;
int j; char b;
{
   char x,y,a[2];
   jp( 4,5,pl ); jp( 6,5,pl );
   jp( 2,5,pl ); jp( 8,5,pl );
   axy( a,enm[j] );
   x = enm[j+1]; y = enm[j+2];
   if ( a[1] == 3 )
   {
      smass( dst,x+1,y  ,b,12,1,8 );
      smass( dst,x  ,y+1,b,12,1,8 );
      smass( dst,x+1,y+2,b,12,1,8 );
      smass( dst,x+2,y  ,b,16,1,8 );
      smass( dst,x+1,y+1,b,16,1,8 );
      smass( dst,x+1,y+2,b,12,1,8 );
   }
   else
   {
      smass( dst,x  ,y  ,b,12,1,8 );
      smass( dst,x+1,y+1,b,12,1,8 );
      smass( dst,x+1,y  ,b,16,1,8 );
      smass( dst,x  ,y+1,b,16,1,8 );
   }
   enm[j] = 0;
}

apr( a,b )
char a,b;
{
   if ( a > b )
   {
      if ( a - b > 8 )
         a++;
      else
         a--;
   }
   if ( a < b )
   {
      if ( b - a > 8 )
         a--;
      else
         a++;
   }
   a += 16; a %= 16;
   return( a );
}

dr( x,y,a,b )
char x,y,a,b;
{
   char c;
   c = 0;
   if ( y > b )
   {
      if ( x == a )
         c = 4;
      if ( x < a )
         c = 2;
      if ( x > a )
         c = 6;
   }
   if ( y < b )
   {
      if ( x == a )
         c = 12;
      if ( x > a )
         c = 10;
      if ( x < a )
         c = 14;
   }
   if ( y == b )
   {
      if ( x < a )
         c = 0;
      if ( x > a )
         c = 8;
   }
   return( c );
}

scit( high )
 int high;
{
 char tmp[20];
 int  i;

 cls2( ' ' );
 for ( i = 22; i != 25; i++ )
 {
 pr( 13,i,':' );
 pr( 24,i,':' );
 }
 strcpy( tmp,"SCORE:" );
 msg( 26,22,tmp );
 strcpy( tmp,"HIGH :" );
 msg( 26,24,tmp );
 itod( high,tmp,5 );
 strcat( tmp,"0" );
 msg( 32,24,tmp );
 strcpy( tmp,"(          )" );
 msg(  2,22,tmp );
 strcpy( tmp,"EN_/ /‾HM)" );
 msg(  2,23,tmp );
 strcpy( tmp,"(          )" );
 msg(  2,24,tmp );
}

open()
{
 char tmp;
    cls2( '-' );
    scr();
    title();
    msg( 14,20,"hit any key" );
    scr();
    tmp = inkey( 2 );
    if ( tmp == 'Q' )
       exit();
}
```

リスト2

```
/* enemy.C */

enemy( enm,eb1,eb2,pl )
char *enm,*eb1,*eb2,*pl;
{
   int i;
   for ( i = 0; i < 128; i += 16 )
      if ( enm[ i ] != 0 )
      {
         if ( enm[ i ] == 1 )
               enmy1( enm,eb1,pl,i );
         if ( enm[ i ] == 2 )
               enmy2( enm,eb2,pl,i );
         if ( enm[ i ] == 3 )
               enmy3( enm,i );
         if ( enm[ i ] == 4 )
               enmy4( enm,eb1,pl,i );
         if ( enm[ i ] == 5 )
               enmy5( enm,eb1,pl,i );
         if ( enm[ i ] == 6 )
               enmy6( enm,eb2,pl,i );
         if ( enm[ i ] == 7 )
               enmy7( enm,eb1,eb2,pl,i );
         if ( enm[ i ] == 8 )
               enmy8( enm,i );
         if ( enm[ i ] == 9 )
               enmy9( enm,i );
      }
}

enmy1( enm,eb1,pl,i )
char *enm,*eb1,*pl;
int i;
{
   char a;
   move( &enm[i] );
   a = 3;
   if ( enm[i+1] > 25 )
   {
      if ( enm[i+9] == 0 )
         enm[i+3] = chhx( enm[i+3],2 );
      else
      {
         enm[i+3] = aprz( enm[i+3],2 );
         a = 4;
      }
   }
   if ( enm[i+3] == 0 && enm[i+9] != 0 )
   {
      enm[i+9]--;
      ensh( pl[1],enm[i+1],enm[i+2],eb1 );
   }
   sp( enm[i+1],enm[i+2],a );
   enck( enm,i );
}

enmy2( enm,eb2,pl,i )
 char *enm,*eb2,*pl;
 int i;
 {
   char a;
   a = 5;
   move( &enm[i] );
   if ( enm[i+1] > 28 )
   {
      if ( enm[i+9] == 0 )
         enm[i+3] = chhx( enm[i+3],2 );
      else
      {
         enm[i+3] = aprz( enm[i+3],2 );
         a = 6;
      }
   }
   if ( enm[i+3] == 0 && enm[i+9] != 0 )
   {
      enm[i+9]--;
      ensh( pl[1],enm[i+1],enm[i+2]-1,eb2 );
   }
   sp( enm[i+1],enm[i+2],a );
   enck( enm,i );
 }

 enmy3( enm,i )
 char *enm;
 int i;
 {
   char a,b;
   move( &enm[i] );
   b = 7;
   if ( enm[i+10] == 0 )
   {
      if ( enm[i+1] > 25 )
```

▶ ショートプロぱーてぃもその(36)，つまり3周年を迎えたのに何も行われませんでした。で，僕からお祝いの言葉「祝！ 3周年。これからも頑張ってください」。

坪田　雅己(17)広島県

```
            enm[i+10] = 1;
      }
      else
      {
         a = enm[i+10];
         if ( a < 7 )
            enm[i+5] -= 2;
         if ( a > 10 )
         {
            enm[i+5] += 2;
            b = 8;
            if ( a == 17 )
               enm[i+10] = 0;
         }
         enm[i+10]++;
      }
      sp( enm[i+1],enm[i+2],b );
      enck( enm,i );
}

enmy4( enm,eb1,pl,i )
char *enm,*eb1,*pl;
int i;
{
   char a,b;
   move( &enm[i] );
   b = 9;
   if ( enm[i+10] == 0 )
   {
      if ( enm[i+1] > 25 )
         enm[i+10] = 1;
   }
   else
   {
      a = enm[i+10];
      if ( a < 7 )
      {
         enm[i+3] -= 2;
         enm[i+10]++;
      }
      if ( a > 7 )
      {
         enm[i+3] = chhx( enm[i+5],2 );
         enm[i+4] = aprz( enm[i+4],3 );
         b = 10;
      }
      if ( a == 7 )
      {
         enm[i+3] = 0;
         chy( enm,pl,i );
         if ( pl[2] == enm[i+2] )
         {
          if ( enm[i+9] != 0 )
          {
           ensh( pl[1],enm[i+1],enm[i+2],eb1 );
           enm[i+9]--;
          }
          else
           enm[i+10] = 8;
         }
      }
   }
   sp( enm[i+1],enm[i+2],b );
   enck( enm,i );
}
enmy5( enm,eb1,pl,i )
char *enm,*eb1,*pl;
int i;
{
   char a,b;
   move( &enm[i] );
   b = 11;
   if ( enm[i+10] == 0 )
   {
      if ( enm[i+1] > 25 )
         enm[i+10] = 1;
   }
   else
   {
      a = enm[i+10];
      if ( a < 14 )
      {
         enm[i+3] = chhx( enm[i+3],-3 );
         enm[i+10]++;
         b = 12;
      }
      if ( a == 14 )
      {
         enm[i+3] = aprz( enm[i+3],3 );
         chy( enm,pl,i );
         if ( enm[i+2] == pl[2] )
         {
          if ( enm[i+9] != 0 )
          {
           ensh( pl[1],enm[i+1],enm[i+2],eb1 );
           enm[i+9]--;
          }
          else
           enm[i+10] = 15;
         }
      }
      if ( a > 14 )
      {
         enm[i+4] = aprz( enm[i+4],2 );
         enm[i+3] = chhx( enm[i+3],-2 );
         b = 12;
      }
   }
   sp( enm[i+1],enm[i+2],b );
   enck( enm,i );
}

enmy6( enm,eb2,pl,i )
char *enm,*eb2,*pl;
int i;
{
   char a,b,c;
   b = enm[i+11];
   a = enm[i+10];
   move( &enm[i] );
   if ( a == 0 )
   {
      enm[i+3] = aprz( enm[i+3],4 );
      enm[i+4] = chhx( enm[i+4],b*2 );
      if ( enm[i+3] == 0 )
         enm[i+10] = 1;
   }
   else
   {
      enm[i+10]++;
      if ( enm[i+10] % 5 == 0 )
      {
         if ( enm[i+9] != 0 )
         {
          ensh( pl[1],enm[i+1],enm[i+2],eb2 );
          enm[i+9]--;
         }
         else
         {
          enm[i+3] = chhx( enm[i+3],3 );
          enm[i+4] = aprz( enm[i+4],2 );
         }
      }
   }
   c = 14;
   if ( enm[i+4] > 0 )
      c = 13;
   if ( enm[i+4] < 0 )
      c = 15;
   sp( enm[i+1],enm[i+2],c );
   enck( enm,i );
}

enmy7( enm,eb1,eb2,pl,i )
char *enm,*eb1,*eb2,*pl;
int i;
{
   char a;
   move( &enm[i] );
   a = 17;
   if ( enm[i+10] == 0 )
   {
      enm[i+3] = aprz( enm[i+3],3 );
      chy( enm,pl,i );
      if ( enm[i+2] == pl[2]
            && enm[i+9] != 0 )
      {
         ensh( pl[1],enm[i+1],enm[i+2],eb1 );
         enm[i+9]--;
      }
      if ( enm[i+9] == 0 )
         enm[i+10] = 1;
   }
   else
   {
      enm[i+4] = aprz( enm[i+4],2 );
      enm[i+3] = chhx( enm[i+3],2 );
      if ( enm[i+1] == 35 )
         ensh( pl[1],enm[i+1],enm[i+2],eb2 );
      a = 16;
   }
   sp( enm[i+1],enm[i+2],a );
   enck( enm,i );
}

enmy8( enm,i )
char *enm;
int i;
{
   move( &enm[i] );
   move( &enm[i] );
   sp( enm[i+1],enm[i+2],18 );
   enck( enm,i );
}

enmy9( enm,i )
char *enm;
int i;
{
   move( &enm[i] );
   move( &enm[i] );
   sp( enm[i+1],enm[i+2],19 );
   enck( enm,i );
}

enck( enm,i )
char *enm;
int  i;
{
   char x;
   x = enm[i+1];
   if ( x < -2 || x > 41 )
   enm[i] = 0;
}

ensh( a,x,y,eb )
char a,x,y,*eb;
{
   int b;
   b = blnk( eb,16,10 );
   if ( b != -1 )
   {
      eb[b  ] = 1;
      eb[b+1] = x;
      eb[b+2] = y;
      eb[b+5] = 16;
   if ( x > a )
      eb[b+3] =-16;
   else
      eb[b+3] = 16;
```

▶ ECCのパソコン甲子園で銀メダルと電子手帳をもらってしまった。次は，パソコンをもらうぞ！　はっはっは……。　有馬　健(17)大阪府

```
    }
}

aprz( a,b )
char a,b;
{
    if ( a > 0 )
    {
        a -= b;
        if ( a < 0 )
            a = 0;
    }
    if ( a < 0 )
    {
        a += b;
        if ( a > 0 )
            a = 0;
    }
    return( a );
}

chy( enm,pl,i )
char *enm,*pl;
int i;
{
    if ( enm[i+2] > pl[2] )
        enm[i+4] = chhx( enm[i+4],-3 );
    if ( enm[i+2] < pl[2] )
        enm[i+4] = chhx( enm[i+4],3 );
}
```

リスト3

```
/* enbl1.C */

 enbl1( eb1 )
 char *eb1;
 {
    char x;
    int i;
    for ( i = 0; i < 160; i += 10 )
    {
        if ( eb1[i] != 0 )
        {
            move( &eb1[i] );
            x = eb1[i+1];
            if ( x < 0 || x > 39 )
                eb1[i] = 0;
            else
                pr( x,eb1[i+2],'-' );
        }
    }
 }

```

リスト4

```
/* enbl2.C */

 enbl2( eb2,pl )
 char *eb2,*pl;
 {
    int i;
    char a,x,y;
    for ( i = 0; i < 160; i += 10 )
    {
        if ( eb2[i] != 0 )
        {
            move( &eb2[i] );
            x = eb2[i+1];
            y = eb2[i+2];
            a = '|';
            if ( eb2[i] == 1 )
            {
                a = '=';
                if ( x - pl[1] <  2
                  && x - pl[1] >= 0 )
                {
                    eb2[i]   = 2;
                    eb2[i+5] = 8;
                }
            }
            if ( eb2[i] == 2 )
            {
                eb2[i+9]++;
                a = '+';
                if ( eb2[i+9] == 2 )
                    {
                        eb2[i] = 3;
                        eb2[i+3] = 0;
                        eb2[i+5] = 16;
                        if ( y > pl[2] )
                            eb2[i+4] =-16;
                        else
                            eb2[i+4] = 16;
                    }
            }
            if ( y < 0 || y > 20 )
                eb2[i] = 0;
            pr( x,y,a );
        }
    }
}

```

リスト5

```
/* plsh.C */

extern blnk();
extern chhx();
extern smass();
extern jp();

 plsh( pl,pbl,dst )
  char *pl,*pbl,*dst;
 {
    int i;
    if ( pl[9] > 2 )
    {
     i = blnk( pbl,22,10 );
     if ( i != -1 )
     {
      pl[9] -= 3;
      pbl[ i ] = 1;
      pbl[ i + 1 ] = pl[1]+2;
      pbl[ i + 2 ] = pl[2];
      pbl[ i + 3 ] = 16;
      pbl[ i + 5 ] = 16;
      pl[3] = chhx( pl[3],-4 );
      smass( dst,pl[1]+2,pl[2]+1,0,16,48,16);
      jp( i%8+1,5,pl );
     }
    }
 }
```

リスト6

```
; indc.ASM

;-------------------------
;   indc ( pl )
;-------------------------

 EXT       ADD

indc::
    POP    BC
    POP    IX
    PUSH   IX
    PUSH   BC ; <-- PREP.
    LD     H,3
    LD     L,22
    CALL   ADD
    LD     A,(IX+9)
    INC    A
    SRL    A
    SRL    A
    CP     9
    JR     C,EPSTEP1
    SUB    8
    LD     B,A
    LD     C,A
    LD     A,":"
    CALL   STRING
    LD     A,8
    SUB    C
    JR     Z,EPSTEP2
    LD     B,A
    LD     A,"."
    CALL   STRING
    JR     EPSTEP2
EPSTEP1:
    LD     B,A
    LD     C,A
    CP     0
    JR     Z,EPSTEP3
    LD     A,"."
    CALL   STRING
EPSTEP3:
    LD     A,8
    SUB    C
    JR     Z,EPSTEP2
    LD     B,A
    LD     A," "
    CALL   STRING
EPSTEP2:
    LD     H,3
    LD     L,24
    CALL   ADD
    LD     A,(IX+10)
    LD     B,A
    LD     C,A
    CP     0
    JR     Z,HPSTEP1
    LD     A,"c"
    CALL   STRING
HPSTEP1:
    LD     A,8
    SUB    C
    JR     Z,HPSTEP2
    LD     B,A
```

▶Oh!XとOh!Xが雨で濡れてくっついた。むりやりはがしたらハゲた。
植地　計友(18)奈良県

```
        LD      A," "
        CALL    STRING
HPSTEP2:
        LD      H,14
        LD      L,22
        CALL    ADD
        LD      A," "
        LD      B,10
        CALL    STRING
        LD      DE,30
        ADD     HL,DE
        LD      B,10
        CALL    STRING
        ADD     HL,DE
        LD      B,10
        CALL    STRING
        PUSH    IX
        POP     HL
        LD      D,H
        LD      E,L
        LD      BC,13
        ADD     HL,BC
        EX      DE,HL
        LD      C,23
        ADD     HL,BC
        EX      DE,HL
        LD      B,10
LOOP1:
        LD      A,(DE)
        ADD     A,(HL)
        CP      35
        JR      C,FLSTEP1
        CP      128
        JR      C,FLSTEP2
        LD      A,0
        LD      (HL),A
        JR      FLSTEP1
FLSTEP2:
        LD      A,35
FLSTEP1:
        LD      (DE),A
        SRL     A
        SRL     A
        CALL    FLPRT
        DEC     (HL)
        INC     HL
        INC     DE
        DJNZ    LOOP1
        RET
STRING:
        LD      (HL),A
        INC     HL
        DJNZ    STRING
        RET
FLPRT:
        PUSH    HL
        PUSH    DE
        PUSH    BC
        LD      C,A
        LD      A,B
        ADD     A,13
        LD      H,A
        CALL    MOD3
        LD      A,24
        SUB     B
        LD      L,A
        CALL    ADD
        EX      DE,HL
        LD      HL,FLTOP
        LD      B,0
        ADD     HL,BC
        LDI
        POP     BC
        POP     DE
        POP     HL
        RET
MOD3:
        PUSH    HL
        LD      B,0
        SLA     C
        LD      HL,MDTOP
        ADD     HL,BC
        LD      B,(HL)
        INC     HL
        LD      C,(HL)
        POP     HL
        RET
FLTOP:
        DM      "_-‾"
MDTOP:
        DB      0,0,0,1,0,2,1,0,1,1,1,2
        DB      2,0,2,1,2,2
```

リスト7

```
/* stit.C */

extern ind( str,a );

stit( str )
 char *str;
 {
   char a,b;
   int  i;
   ind( str,120 );
   for( i = 0; i < 120; i +=10 )
   {
     str[ i + 1 ] = rnd( 39 );
     str[ i + 2 ] = rnd( 21 );
     if ( i < 40 )
     {
          a = 16 ; b = '。';
     }
     if ( i > 39 && i < 80 )
     {
          a = 12 ; b = '·';
     }
     if ( i > 79 )
     {
          a = 8  ; b = '.';
     }
     str[ i + 5 ] = a;
     str[ i + 9 ] = b;
     str[ i + 3 ] =-16;
   }
 }
```

リスト8

```
/* smass.C */

extern blnk();
extern dx();
extern dy();

smass( dst,a,b,c,d,e,f )
 char *dst,a,b,c,d,e,f;
 {
    int i;
    i = blnk( dst,64,10 );
    if ( i != -1 )
    {
       dst[ i ] = e;
       dst[ i + 1 ] = a;
       dst[ i + 2 ] = b;
       dst[ i + 3 ] = dx( c );
       dst[ i + 4 ] = dy( c );
       dst[ i + 5 ] = d;
       dst[ i + 9 ] = f;
    }
 }
```

リスト9

```
/* chhx.C */

chhx( a,b )
 char a,b;
 {
    a += b;
    if ( a > 16 )
       a = 16;
    if ( a < -16 )
        a = -16;
    return( a );
 }
```

リスト10

```
/* jp.C */

 jp( a,b,pl )
 char a,b,*pl;
 {
    a %= 10;
    pl[13+a] = ckj( pl[13+a] , b   );
    pl[12+a] = ckj( pl[12+a] , b/2 );
    pl[14+a] = ckj( pl[14+a] , b/2 );
 }
 ckj( a,b )
 char a,b;
 {
    a += b;
    if ( a > 8 )
     a = 8;
    return( a );
 }
```

リスト11

```
/* ind.C */

 ind( add , c )
  char *add;
  int  c;
 {
  int i;
  for( i = 0; i < c; i++ )
     add[i] = 0;
 }
```

▶ ハードディスクが欲しい。メモリも欲しい。うちのPC-9801が父のものでなかったら，とっとと売ってしまうのだが……。 河田　渉(20)北海道

リスト12

```
; title.ASM

;--------------------------
;    title ( )
;--------------------------

 EXT ADD

title::
 LD   H,5
 LD   L,4
 CALL ADD
 LD   B,14
 LD   DE,TTTOP
LOOP:
 PUSH BC
 LD   C,4
LOOP2:
 LD   B,8
 LD   A,(DE)
 INC  DE
LOOP3:
 RLA
 JR   NC,STEP1
 LD   (HL),"H"
STEP1:
 INC  HL
 DJNZ LOOP3
 DEC  C
 JR   NZ,LOOP2
 LD   BC,8
 ADD  HL,BC
 POP  BC
 DJNZ LOOP
 RET

TTTOP:
 DB   $74,$74,$B1,$DC,$84,$46,$A9,$12
 DB   $64,$75,$A5,$DC,$14,$44,$A5,$12
 DB   $E7,$74,$B9,$D2,$00,$00,$00,$00
 DB   $E3,$9C,$73,$80,$E3,$9C,$73,$80
 DB   $E3,$9C,$73,$80,$FF,$9C,$73,$80
 DB   $E3,$9C,$73,$80,$E3,$9C,$73,$80
 DB   $E3,$9C,$73,$80,$E3,$8F,$E3,$FE


```

リスト13

```
; enorg.ASM

;--------------------------
;    enorg ( d )
;--------------------------

     EXT    blnk

enorg::
     POP    BC
     POP    DE
     PUSH   DE
     PUSH   BC ; <-- PREP.
     LD     A,0
     CP     D
     JR     NZ,NONIT
; INITIALIZE
     LD     HL,APTOP
     LD     (APADD),HL
     LD     A,E
     LD     (COUNT),A
     RET
; TAKING DATA OUT FROM AP. TAB.
NONIT:
     LD     HL,-1
     LD     A,(COUNT)
     CP     255
     RET    Z
     LD     HL,0
     DEC    A
     LD     (COUNT),A
     RET    NZ
     LD     HL,(APADD)
     LD     A,(HL)
     INC    HL
     LD     (COUNT),A
     LD     A,(HL)
     INC    HL
     LD     B,(HL)
     INC    HL
     LD     C,(HL)
     INC    HL
     LD     (APADD),HL
     PUSH   BC
     LD     B,A
     ADD    A,A
     ADD    A,B
     ADD    A,A
     LD     C,A
     LD     B,0
     LD     HL,TBTOP
     ADD    HL,BC
     PUSH   HL
     PUSH   DE
     LD     DE,16
     PUSH   DE
     PUSH   DE
     CALL   blnk
     POP    BC
     POP    BC
     POP    IX
     POP    DE
     POP    BC
     INC    H
     RET    C
     DEC    H
     EX     DE,HL
     ADD    IX,DE
     LD     (IX+1),B
     LD     (IX+2),C
     LD     A,(HL)
     INC    HL
     LD     (IX+0),A
     LD     A,(HL)
     INC    HL
     LD     (IX+3),A
     LD     A,(HL)
     INC    HL
     LD     (IX+4),A
     LD     A,(HL)
     INC    HL
     LD     (IX+5),A
     LD     A,(HL)
     INC    HL
     LD     (IX+9),A
     LD     A,(HL)
     INC    HL
     LD     (IX+11),A
     LD     A,(IX+0)
     LD     B,0
     LD     C,A
     DEC    BC
     LD     HL,STTOP
     ADD    HL,BC
     LD     A,(HL)
     LD     (IX+12),A
     RET
COUNT:
     DB     0
APADD:
     DW     0
     ;      k,dx,dy, s,b,dr, k,dx,dy, s,b,dr
TBTOP:
     DB     1,16, 0,10,0, 0, 1,16, 0,16,0, 0
     DB     1,16, 0,10,1, 0, 1,16, 0,16,1, 0
     DB     2,16, 0,10,1, 0, 2,16, 0,16,1, 0
     DB     2,16, 0,10,3, 0, 2,16, 0,16,3, 0
     DB     3,16, 0,16,0, 0
     DB     4,16, 0,16,0, 0, 5,16, 0,16,0, 0
     DB     4,16, 0,16,1, 0, 5,16, 0,16,1, 0
     DB     4,16, 0,16,3, 0, 5,16, 0,16,3, 0
     DB     6,16, 0,16,0,-1, 6,16, 0,16,0, 1
     DB     6,16, 0,16,1,-1, 6,16, 0,16,1, 1
     DB     6,16, 0,16,3,-1, 6,16, 0,16,3, 1
     DB     7, 8, 0,16,0, 0, 7,12, 0,16,1, 0
     DB     7,12, 0,16,3, 0,8,-16, 0,12,0, 0
     DB    8,-16, 0,16,0, 0,9,-16, 0,16,0, 0
     DB    8,-16,-6,12,0, 0,8,-16,-6,16,2, 0
STTOP:
     DB     4,5,100,3,2,6,10,12,14

     ;  tm,kd,ex,ey, tm,kd,ex,ey, tm,kd,ex,ey
APTOP:
     DB  1, 0,-1, 5, 48, 0,-1,15, 64, 1,-1,10
     DB 32, 1,-1,12, 32, 1,-1, 6, 32, 1,-1, 9
     DB 24, 1,-1,17, 16, 2,-1,14, 32, 2,-1, 3
     DB 16, 3,-1, 6, 84, 2,-1,11
     DB 48, 5,-1,13, 32, 4,-1, 6, 16, 1,-1, 6
     DB 16, 1,-1,16, 64, 3,-1, 3, 64, 7,-1,19
     DB 48, 1,-1, 5,  1, 6,-1,18, 76, 6,-1,15
     DB  1, 7,-1, 4, 48, 7,-1, 6, 24, 2,-1,16
     DB 76, 3,-1,18, 16, 9,-1, 2, 64, 9,-1, 4
     DB  1, 5,-1, 3, 64, 5,-1, 6, 16, 1,-1,10
     DB 32, 1,-1,13, 52, 2,-1, 4, 16, 9,-1,15
     DB 76,11,-1,15, 16, 6,-1, 5,  1, 6,-1, 2
     DB 16,11,-1,10, 76,11,-1,13, 32, 1,-1, 5
     DB 24, 1,-1,18, 24, 0,-1, 8,  8, 1,-1,15
     DB 16, 1,-1,10, 92, 0,-1,13
     DB 16, 6,-1, 9, 48, 6,-1,11, 16,10,-1,18
     DB 86,10,-1, 2, 32, 2,-1,16, 16, 3,-1,14
     DB 92, 3,-1,12, 16, 8,-1, 2, 16, 8,-1, 5
     DB 48, 7,-1,10, 16, 7,-1,15, 42, 7,-1,18
     DB 12, 1,-1,16, 32, 1,-1,18, 24, 7,-1, 8
     DB 38, 7,-1, 2, 38,12,-1, 8,  1, 8,-1, 2
     DB 12,13, 1, 3, 12, 8,-1, 5, 24, 8,-1, 7
     DB 12, 8,-1,12, 16, 7,-1,10, 12, 5,-1,12
     DB 16, 8,-1,14, 32, 8,-1,18, 12,13,-1,18
     DB 24,13,-1,14,  1, 8,-1, 2,  1,14, 1, 3
     DB  1, 8,-1, 5, 32,14, 1, 6,  8, 7,-1,15
     DB 16, 7,-1,19, 32, 7,-1,13,  8, 7,-1, 5
     DB 16, 7,-1, 3, 24, 7,-1, 1,  1, 8,-1,15
     DB  1,13, 1,16,  1, 8,-1,18, 36,13, 1,19
     DB 16, 1,-1,10, 32, 3,-1, 8, 48, 1,-1,15
     DB 92, 1,-1, 5, 32,26,39,10, 24,26,39,15
     DB 12,26,39, 5,  8,26,39, 8, 12,26,39,16
     DB  8,26,39,12,  4,26,39,10,  8,26,39,18
     DB  8,26,39, 5,  4,26,39,14, 92,26,39,10
     DB  1, 8,-1, 5,  8,11, 1, 6,  1, 8,-1, 9
     DB 32,14, 1,10,  1, 8,-1,12,  8,13, 1,13
     DB  1, 8,-1,16, 24,13, 1,17,  4,26,39, 5
     DB  4, 7,-1,14, 28,26,39, 8,  1,26,39,13
     DB 48, 7,-1,16,  1, 8,-1, 2,  1,14, 1, 3
     DB  1, 8,-1, 6,  1,14, 1, 7,  1, 8,-1,11
     DB 32,14, 1,12, 12, 3,-1,18, 32, 3,-1,15
     DB 12, 6,-1, 4, 24, 6,-1, 7,  1, 8,-1,17
     DB  8,13, 1,18,  1, 8,-1,14, 16,13, 1,15
     DB  1,26,39, 3,  4,26,39, 9, 32,26,39, 6
     DB 16, 7,-1,14, 32, 7,-1,16, 24,26,39,12
     DB 32,26,39,17, 24, 3,-1, 5, 48, 3,-1, 2
     DB 32, 1,-1,15, 92, 1,-1,10
     DB 32,24,39, 5, 24,24,39,12,  8,24,39, 3
     DB 16,24,39,18, 24,24,39,15,  8,25,39,10
     DB 16,24,39, 5, 12,25,39,12,  4,24,39, 2
     DB 24,25,39,16, 16,25,39, 8,  4,24,39,15
     DB 12,24,39, 6,  4,25,39,14, 16,25,39, 8
     DB  6,24,39, 4, 16, 5,-1,18, 12,25,39,16
```

▶ハードディスクが入荷しない……。ロジテックのLHD-FM200Eである。いつになったら、快適な環境を満喫できるのだろうか。フロッピーを抜き差しするたびに友人をうらやましく思う。早く来い来い，ハードディスク……。　新井　誠治(21)北海道

```
        DB 16, 1,-1,10,  8, 1,-1,13, 16,25,39, 5
        DB  8,24,39,15, 12,25,39, 8,  4,11,-1,18
        DB 10,25,39, 3, 14,25,39,12,  8,11,-1,15
        DB  8, 7,-1, 3,  8, 7,-1, 5, 12,25,39,13
        DB  4,24,39, 4, 12,25,39, 8,  8,24,39,15
        DB 16,24,39, 3, 14,24,39, 7, 16,24,39,18
        DB 16,24,39,12,  8,24,39,10, 10,25,39, 5
        DB  1, 8,-1,14,  1,13, 1,15,  1, 8,-1,17
        DB  8,13, 1,18, 12,24,39, 8,  8,24,39, 3
        DB 16,24,39,10, 12,24,39,15, 12,24,39, 2
        DB 12,24,39,18, 64,24,39, 8,  1,24,39, 9
        DB  1,24,39,13,  1,24,39, 5,  1,24,39, 1
        DB 16,24,39,17, 12,24,39, 8, 32,24,39,15
        DB  4,26,39,15,  8,26,39,12, 52,25,39, 5
        DB  8, 6,-1,17, 24, 7,-1,14, 12, 1,-1, 3
        DB  1, 8,-1, 5, 12,14, 1, 6,  1, 8,-1, 8
        DB 24,14, 1, 9, 32, 1,-1,16, 12, 4,-1,18
        DB 32, 4,-1,16, 24, 6,-1, 2, 32, 7,-1,17
        DB  8,26,39, 8, 24,26,39,12, 48,13,-1,18
        DB  1, 8,-1, 3,  8,13, 1, 4,  1, 8,-1, 6
        DB 32,13, 1, 7, 16,21,-1,16, 24,21,-1,18
        DB 16, 1,-1,12, 48, 1,-1, 7,  1, 8,-1,12
        DB 16,14, 1,13,  1, 8,-1,16, 16,14, 1,17
        DB 16, 1,-1, 8, 28, 1,-1, 3, 20,22,-1, 6
        DB 48,22,-1,10, 20, 3,-1,10, 32,22,-1,15
        DB  8,26,39, 8, 12,26,39,12, 16,22,-1, 2
        DB 24,23,-1, 5, 10,26,39,10, 10,26,39, 6
        DB  1, 8,-1,10, 18,14, 1,11,  1, 8,-1, 7
        DB 32,14, 1, 8, 12,23,-1,18, 48,23,-1,14
        DB 24,23,-1, 3, 12, 1,-1, 5, 28,13,-1,16
        DB 32,23,-1,12, 12,23,-1,10,  1,26,39, 5
        DB 16,23,-1, 8, 32,26,39,17, 32, 1,-1,10
        DB 32, 1,-1,16, 48, 7,-1, 5
        DB 32,24,39, 5, 24,24,39,12,  8,24,39, 3
        DB 16,24,39,18, 24,24,39,15,  8,25,39,10
        DB 16,24,39, 5, 12,25,39,12,  4,24,39, 2
        DB 24,25,39,16, 16,25,39, 8,  4,24,39,15
        DB 12,24,39, 6,  4,25,39,14, 16,25,39, 8
        DB  6,24,39, 4, 16,17,20,21, 12,25,39,16
        DB 16,17,18,21,  8,18,32,-1, 16,25,39, 5
        DB  8,24,39,15, 12,25,39, 8,  4,18,18,-1
        DB 10,25,39, 3, 14,25,39,12,  8,17,-1,25
        DB  4,24,39, 4, 12,25,39, 8,  8,24,39,15
        DB 16,24,39, 3, 14,24,39, 7, 16,24,39,18
        DB 16,24,39,12,  8,24,39,10, 10,25,39, 5
        DB  1,19,25,21, 16,25,39,15, 12,20,25,-1
        DB  8,20,25,-1, 12,24,39, 8,  8,24,39, 3
        DB 16,24,39,18, 24,24,39,15,  8,25,39,10
        DB 32,24,39, 5, 24,24,39,12,  8,24,39, 3
        DB 16,24,39, 5, 12,25,39,12,  4,24,39, 2
        DB 24,25,39,16, 16,25,39, 8,  4,24,39,15
        DB 12,25,39, 6,  4,25,39,14, 16,25,39, 8
        DB  6,25,39, 4, 16,19,20,21, 12,25,39,16
        DB 16,25,39,10,  8,20,32,-1, 16,25,39, 5
        DB  8,25,39,15, 12,25,39, 8,  4,20,18,-1
        DB  6,25,39, 4, 16,19,20,21, 12,25,39,16
        DB 16,19,18,21,  8,20,32,-1, 16,25,39, 5
        DB 10,25,39, 3, 14,25,39,12,  8,17,-1,25
        DB  1,19,25,21, 16,25,39,15, 12,20,25,-1
        DB  8,25,29, 3, 12,25,39, 8,  8,25,39, 3
        DB 16,25,39,18, 24,25,39,12,  3,25,39, 6
        DB 32,25,39, 5, 24,25,39,12,  8,25,39, 3
        DB  1, 8,-1, 6,  1,14, 1, 7,  1, 8,-1,11
        DB 32,14, 1,12, 12, 3,-1,18, 32, 3,-1,15
        DB 12, 6,-1, 4, 24, 6,-1, 7,  1, 8,-1,17
        DB  8,13, 1,18,  1, 8,-1,14, 16,13, 1,15
        DB  1,26,39, 3,  4,26,39, 9, 32,26,39, 6
        DB 16, 7,-1,14, 32, 7,-1,16, 24,26,39,12
        DB 32,26,39,17, 24, 3,-1, 5, 48, 3,-1, 2
        DB 32, 1,-1,15, 92, 1,-1,10
        DB 12,26,39, 5,  8,26,39, 8, 12,26,39,16
        DB  8,26,39,12,  4,26,39,10,  8,26,39,18
        DB  8,26,39, 5,  4,26,39,14, 92,26,39,10
        DB  1, 8,-1, 5,  8,11, 1, 6,  1, 8,-1, 9
        DB 32,14, 1,10,  1, 8,-1,12,  8,13, 1,13
        DB  1, 8,-1,16, 24,13, 1,17,  4,26,39, 5
        DB  4, 7,-1,14, 28,26,39, 8,  1,26,39,13
        DB 48, 7,-1,16,  1, 8,-1, 2, 23,14, 1, 3
        DB 16, 1,-1, 8, 28, 1,-1, 3, 20,22,-1, 6
        DB 48,22,-1,10, 20, 3,-1,10, 32,22,-1,15
        DB  8,26,39, 8, 12,26,39,12, 16,22,-1, 2
        DB 24,23,-1, 5, 10,26,39,10, 10,26,39, 6
        DB  1, 8,-1,10, 18,14, 1,11,  1, 8,-1, 7
        DB 32,14, 1, 8, 12,23,-1,18, 48,23,-1,14
        DB 24,23,-1, 3, 12, 1,-1, 5, 28,13,-1,16
        DB 32,23,-1,12, 12,23,-1,10,  1,26,39, 5
        DB 16,23,-1, 8, 32,26,39,17, 32, 1,-1,10
        DB 32, 1,-1,16, 48, 7,-1, 5, 255
```

リスト14

```
; sp.ASM

;-------------------------
;   sp(x,y,n)
;-------------------------

        EXT     PRN

sp::
        INC     SP
        POP     DE
        LD      A,D
        EXX
        POP     BC
        LD      L,B
        POP     DE
        LD      H,D
        PUSH    DE
        PUSH    BC
        PUSH    HL
        EXX
        POP     BC
        PUSH    DE
        DEC     SP              ; <-- PREP.
        ADD     A,A
        LD      D,0
        LD      E,A
        LD      HL,CHTOP
        ADD     HL,DE
        LD      E,(HL)
        INC     HL
        LD      D,(HL)
        LD      H,B
        LD      L,C
        CALL    PRN
        RET

CHTOP:
        DW      PL1,PL2,PL3,E11,E12
        DW      E21,E22,E31,E32,E41,E42,E51,E52
        DW      E61,E62,E63,E71,E72,E81,E91

PL1:
        DB      3,2
        DM      "^-_/• `"
PL2:
        DB      3,2
        DM      "--_/• `"
PL3:
        DB      3,2
        DM      "^-_¯• `"
E11:
        DB      3,2
        DM      ">cO/`@"
E12:
        DB      3,2
        DM      "=cO@)`"
E21:
        DB      4,2
        DM      ").=o`O¯@"
E22:
        DB      3,2
        DM      ",=o)O¯"
E31:
        DB      5,2
        DM      ")ee_@/@@[)"
E32:
        DB      5,2
        DM      ")ee_@¯@@[)"
E41:
        DB      2,2
        DM      "=o/`"
E42:
        DB      2,2
        DM      ")o¯`"
E51:
        DB      2,2
        DM      "cx¯("
E52:
        DB      2,2
        DM      "cx¯="
E61:
        DB      3,3
        DM      "@@!@*^¯^x"
E62:
        DB      3,3
        DM      "^_@* +/¯@"
E63:
        DB      3,3
        DM      "_/x@*/@@!"
E71:
        DB      3,2
        DM      "`eo)¯@"
E72:
        DB      3,2
        DM      "`eo@)¯"
E81:
        DB      3,3
        DM      "@_@( )`-'"
E91:
        DB      5,2
        DM      "@_ee(()@@¯"
```

リスト15

```
; scr.ASM

;-------------------------
;   scr()
;-------------------------

LOC             EQU     201EH
PRINT           EQU     1FF4H

SCADD           EQU     $C000

scr::

        LD      HL,SCADD+7E6H
        LD      BC,SCADD+3E6H
        LD      DE,1826H
LOOP:
        LD      A,(BC)
        CP      (HL)
        JR      Z,STEPSC1
        LD      (HL),A
        EX      DE,HL
        CALL    LOC
        LD      A,(BC)
```


▶う〜む。9月号の表紙の人って，やっぱり男なんでしょうか。よく見ると，○○のような物体がありますからねぇ〜。　榎本　和則(17)大阪府

```
        CALL  PRINT
        EX    DE,HL
STEPSC1:
        DEC   BC
        DEC   HL
        DEC   E
        LD    A,255
        CP    E
        JR    NZ,LOOP
        LD    E,39
        DEC   D
        CP    D
        JR    NZ,LOOP
        RET
```

リスト16

```
; pr.ASM

;--------------------------
;    pr(x,y,n)
;--------------------------

        EXT   ADD

pr::
        INC   SP
        POP   DE
        LD    A,D
        EXX
        POP   BC
        LD    L,B
        POP   DE
        LD    H,D
        PUSH  DE
        PUSH  BC
        PUSH  HL
        EXX
        POP   HL
        PUSH  DE
        DEC   SP            ; <-- PREP.
        LD    B,A
        CALL  ADD
        RET   C
        LD    (HL),B
        RET
```

リスト17

```
; msg.ASM

;--------------------------
;    msg(x,y,d)
;--------------------------

        EXT   ADD

msg::
        POP   BC
        POP   DE
        EXX
        POP   BC
        LD    L,C
        POP   DE
        LD    H,E
        PUSH  DE
        PUSH  BC
        PUSH  HL
        EXX
        POP   HL
        PUSH  DE
        PUSH  BC            ; <-- PREP.
        CALL  ADD
        LD    A,0
        EX    DE,HL
LOOP:
        CP    (HL)
        JR    Z,SKIP
        LDI
        JR    LOOP
SKIP:
        RET
```

リスト18

```
; axy.ASM

;--------------------------
;    axy( d,n )
;--------------------------

axy::
        POP   BC
        POP   DE
        POP   HL
        PUSH  HL
        PUSH  DE
        PUSH  BC ; <-- PREP.
        LD    A,E
        DEC   A
        ADD   A,A
        LD    D,0
        LD    E,A
        PUSH  HL
        LD    HL,XYTOP
        ADD   HL,DE
        POP   DE
        LDI
        LDI
        RET
XYTOP:
        DB    3,2,3,2,5,2,2,2,2,2
        DB    3,3,3,2,3,3,5,2
```

リスト19

```
; dx.ASM

;--------------------------
;    dx(n)
;--------------------------

dx::
        INC   SP
        POP   DE
        PUSH  DE
        DEC   SP  ; <-- PREP.
        LD    E,D
        LD    D,0
        LD    HL,DXTAB
        ADD   HL,DE
        LD    D,(HL)
        LD    H,0
        LD    L,D
        RET
DXTAB:
        DB     16, 15, 11,  6
        DB      0, -6,-11,-15
        DB    -16,-15,-11, -6
        DB      0,  6, 11, 15
```

リスト20

```
; dy.ASM

;--------------------------
;    dy(n)
;--------------------------

dy::
        INC   SP
        POP   DE
        PUSH  DE
        DEC   SP  ; <-- PREP.
        LD    E,D
        LD    D,0
        LD    HL,DYTAB
        ADD   HL,DE
        LD    D,(HL)
        LD    H,0
        LD    L,D
        RET
DYTAB:
        DB      0, -6,-11,-15
        DB    -16,-15,-11, -6
        DB      0,  6, 11, 15
        DB     16, 15, 11,  6
```

リスト21

```
; strs.ASM

;--------------------------
;    strs( d )
;--------------------------

        EXT   ADD
        EXT   move

strs::
        POP   BC
        POP   IX
        PUSH  IX
        PUSH  BC  ; <-- PREP.
        LD    B,12
LOOP:
        PUSH  BC
        PUSH  IX
        CALL  move
        POP   IX
        POP   BC
        LD    H,(IX+1)
        LD    L,(IX+2)
        CALL  ADD
        JR    C,SKIP
        LD    A,(IX+9)
        LD    (HL),A
        JR    SKIP2
SKIP:
        LD    (IX+1),39
SKIP2:
        LD    DE,10
        ADD   IX,DE
        DJNZ  LOOP
        RET
```

▶ムムム。電脳倶楽部がディスクを増やすにあたって，まさか月２回刊を考えていたとは……。Oh!Xも月２回刊にしては……。私なら金のある限り買い続けるぞ！

金子　聡史(18)千葉県

リスト22

```
; plbul.ASM

;-------------------------
;   plbul( d )
;-------------------------

    EXT   ADD

plbul::
    POP   BC
    POP   IX
    PUSH  IX
    PUSH  BC  ; <-- PREP.
    LD    B,22
LOOP:
    LD    A,(IX+0)
    CP    0
    JR    Z,SKIP2
    INC   (IX+1)
    INC   (IX+1)
    LD    H,(IX+1)
    LD    A,39
    CP    H
    JR    NC,SKIP1
    LD    (IX+0),0
    JR    SKIP2
SKIP1:
    LD    L,(IX+2)
    CALL  ADD
    LD    A,"_"
    LD    (HL),A
SKIP2:
    LD    DE,10
    ADD   IX,DE
    DJNZ  LOOP
    RET
```

リスト23

```
; blnk.ASM

;-------------------------
;   blnk ( d,a,b )
;-------------------------

blnk::
    INC   SP
    INC   SP
    POP   BC
    POP   DE
    POP   HL
    PUSH  HL
    PUSH  DE
    PUSH  BC
    DEC   SP
    DEC   SP ; <-- PREP.
    LD    B,E
    LD    DE,0
LOOP1:
    LD    A,(HL)
    CP    0
    JR    Z,FOUND
    LD    A,B
    LD    B,0
    ADD   HL,BC
    EX    DE,HL
    ADD   HL,BC
    EX    DE,HL
    LD    B,A
    DJNZ  LOOP1
    LD    HL,-1
    RET
FOUND:
    LD    B,C
    LD    A,0
LOOP2:
    LD    (HL),A
    INC   HL
    DJNZ  LOOP2
    EX    DE,HL
    RET

```

リスト24

```
;  move.ASM

;-------------------------
;       move(d)
;-------------------------

move::

    POP   BC
    POP   IX          ; <-- ADD
    PUSH  IX
    PUSH  BC
    LD    A,(IX+5)  ; Speed
    ADD   A,(IX+8)
    LD    B,A
    AND   15
    LD    (IX+8),A
    CP    B
    JR    Z,RET
    LD    A,(IX+3)
    CP    127         ; Chk -
    JR    NC,XMINUS
    ADD   A,(IX+6)  ; dX > 0
    LD    B,A
    AND   15
    LD    (IX+6),A
    CP    B
    JR    Z,SKPY
    INC   (IX+1)
    JR    SKPY
XMINUS:
    XOR   255         ; X < 0
    LD    B,A
    INC   B
    LD    A,(IX+6)
    ADD   A,B
    LD    C,A
    AND   15
    LD    (IX+6),A
    CP    C
    JR    Z,SKPY
    DEC   (IX+1)
SKPY:                 ; Y Axis
    LD    A,(IX+4)
    CP    127         ; Chk -
    JR    NC,YMINUS
    ADD   A,(IX+7)  ; dX > 0
    LD    B,A
    AND   15
    LD    (IX+7),A
    CP    B
    JR    Z,RET
    INC   (IX+2)
    JR    RET
YMINUS:
    XOR   255         ; Y < 0
    LD    B,A
    INC   B
    LD    A,(IX+7)
    ADD   A,B
    LD    C,A
    AND   15
    LD    (IX+7),A
    CP    C
    JR    Z,RET
    DEC   (IX+2)
    OR    A
RET:
    RET
```

リスト25

```
; cls2.ASM

;-------------------------
;   cls2( n )
;-------------------------

SCADD       EQU $C000

cls2::

    INC   SP
    POP   BC
    PUSH  BC
    DEC   SP
    LD    A,0
    CP    B
    JR    Z,CLSKIP
    LD    DE,3E8H
    LD    HL,SCADD
LOOP:
    LD    (HL),B
    INC   HL
    DEC   DE
    LD    A,E
    OR    D
    JR    NZ,LOOP
    RET
;-------------------------
;   cls only upper area
;-------------------------

CLSKIP:
    LD    DE,370H
    LD    HL,SCADD
    LD    B," "
LOOP2:
    LD    (HL),B
    INC   HL
    DEC   DE
    LD    A,E
    OR    D
    JR    NZ,LOOP2
    RET

```

▶裏表紙が奇麗になった。ところで，感性専用って？　兼　英樹(18)大阪府

リスト26

```
; smoke.ASM

;-------------------------
;   smoke( d )
;-------------------------

    EXT   ADD
    EXT   move

smoke::
    POP   BC
    POP   IX
    PUSH  IX
    PUSH  BC   ; <-- PREP.
    LD    B,64
LOOP:
    LD    A,(IX+0)
    CP    0
    JR    Z,SKIP
    PUSH  BC
    PUSH  IX
    CALL  move
    POP   IX
    POP   BC
    LD    H,(IX+1)
    LD    L,(IX+2)
    LD    A,21
    CP    L
    JR    C,SKIP2
    CALL  ADD
    JR    C,SKIP2
    LD    A,(IX+0)
    ADD   A,(IX+9)
    LD    (IX+0),A
    AND   $F0
    CP    $70
    JR    NC,SKIP2
    EX    DE,HL
    SRL   A
    SRL   A
    SRL   A
    SRL   A
    LD    HL,SMTOP
    PUSH  BC
    LD    C,A
    LD    B,0
    ADD   HL,BC
    POP   BC
    LD    A,(HL)
    LD    (DE),A
    JR    SKIP
SKIP2:
    LD    (IX+0),0
SKIP:
    LD    DE,10
    ADD   IX,DE
    DJNZ  LOOP
    RET

SMTOP:
    DM    ".。coc。."
```

リスト27

```
; PRN.ASM

;-------------------------
;   PRN
;-------------------------

    EXT   ADD

PRN::

    PUSH  HL
    PUSH  DE
    PUSH  BC
    LD    A,(DE)
    LD    B,A
    INC   DE
    LD    A,(DE)
    LD    C,A
    INC   DE
LOOP1:
    PUSH  BC
    PUSH  DE
    PUSH  HL
    LD    A,21
    CP    L
    JR    C,HSKIP
    LD    A,39
    CP    H
    JR    NC,LSKIP1
    LD    A,B
    ADD   A,H
    JR    Z,HSKIP
    CP    40
    JR    NC,HSKIP
    LD    H,A
    LD    C,A
    LD    A,B
    SUB   C
    LD    C,A
    LD    B,0
    EX    DE,HL
    ADD   HL,BC
    EX    DE,HL
    LD    B,H
    LD    H,0
    JR    LSKIP2
LSKIP1:
    LD    A,B
    ADD   A,H
    CP    40
    JR    C,LSKIP2
    SUB   40
    LD    C,A
    LD    A,B
    SUB   C
    LD    B,A
LSKIP2:
    CALL  ADD
    EX    DE,HL
    LD    A,40H
    LD    C,255
LOOP2:
    CP    (HL)
    JR    Z,SKIPL2
    LDI
    DJNZ  LOOP2
    JR    HSKIP
SKIPL2:
    INC   DE
    INC   HL
    DJNZ  LOOP2
HSKIP:
    POP   DE
    INC   E
    POP   HL
    POP   BC
    LD    A,C
    LD    C,B
    LD    B,0
    ADD   HL,BC
    EX    DE,HL
    LD    B,C
    LD    C,A
    DEC   C
    JR    NZ,LOOP1
    POP   BC
    POP   DE
    POP   HL
    RET
```

リスト28

```
; ADD.ASM

;-------------------------
;   ?ADD
;-------------------------

SCADD       EQU $C000

ADD::

    PUSH  DE
    LD    A,39
    CP    H
    JR    C,ERR
    LD    A,24
    CP    L
    JR    C,ERR
    LD    A,H
    LD    H,0
    LD    D,H
    LD    E,L
    ADD   HL,HL
    ADD   HL,HL
    ADD   HL,DE
    ADD   HL,HL
    ADD   HL,HL
    ADD   HL,HL
    LD    D,0
    LD    E,A
    ADD   HL,DE
    LD    DE,SCADD
    ADD   HL,DE
    OR    A
    JR    NONERR
ERR:
    LD    HL,$C400
    SCF
NONERR:
    POP   DE
    RET
```

▶デモを見て，まずカンドーした。ファイナルファイトを買って，プレイして超カンドーした。でも，次の日，X68000のディスクドライブがいかれて，プレイできなくなった。悲しい……。
中川　直之(20)大阪府

KYOKO
ANOTHER
IN
CG
WORLD
かわいく描いてニャ
おまかせ おまかせ
ひらひら
んでんでこんなカンジ…
3次元だとにできるはずないん…
ひらりん
OK OK OK
Let's render
わくわくわく
わくわくわくわく
わく
わく
わくわくわく
がちょーん!

今回のCGデータ

総物体数　418

光源　3

1280×1024ピクセル

1670万色フルカラーを

4×5ポジで出力

使用ソフトは,

サイクロン, C-TRACE

ポリゴンデータ作成に

Z's TRIPHONY

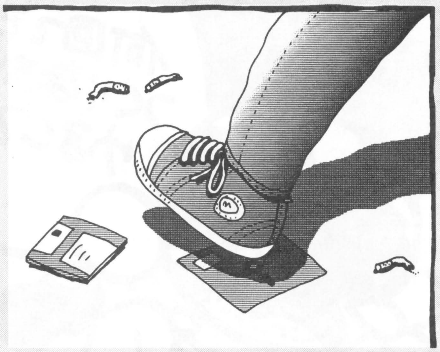
マッピングデータ作成に

Z's STAFF PRO-68K, MATIER

X-OVER・NIGHT
(クロスオーバーナイト)

［第27話］

パソコンドリームは終わった



TAKAHARA HIDEKI 高原 秀己

パソコンの仕事に携わっていさえすれば，いつの日にかたくさんの収入を得て，時代の寵児になれる，と誰もが信じて疑わない時代があった。

そこまでオーバーではないにせよ，何かのはずみで自分も突然，著名人の片隅に交じってしまうのではないか？　パソコン関連企業に勤めている人ならば，誰しも一度はこんなことをふと妄想したことがあるはずだ。

そんな1980年代も遠い過去になったと実感しつつあった8月中旬のある日，日刊工業新聞に大々的に掲載された記事は，こうしたパソコンドリーミングの時代が完全に終わってしまったことを実感させたかのような気がする。

「アスキー，キヤノンに支援要請」――。

事情通の人は「いよいよか」と思ったらしいが，大方の人はこの記事に少なからず驚いたはず。

なにしろ，アスキーといえば，社長があの西和彦氏。本誌出版元であるソフトバンクの孫正義社長と並んで，わが国におけるパソコンドリーミングの象徴的な人物だ。パソコン雑誌を定着させ，マイクロソフトBASICとMS-DOSをわが国に普及させた実績もさることながら，やはり圧巻だったのが1983年のMSXプロジェクトのときに見せた「大活躍」だ。わずか26歳(当時)の人物がほぼすべての家電メーカーを手玉に取り，大手各社重役を従えた華々しさは，年功序列，大手優先の感覚がすべてである日本のビジネス社会をあっといわせ，一躍，時の人となった。

ビル・ゲイツ，マイクロソフト会長，スティーブン・ジョブス，アップルコンピュータ元会長らの活躍と西氏が対比されることで，パソコンという正体不明，未知数無限大のビジネスをもってすれば，日本でもアメリカ並みのドリーミング・サクセスストーリーが可能なのか，と世間に思い知らしめた点も大きかった。

その西氏も，いつしかアスキーの副社長から社長に昇格している。株式も店頭公開し，アスキー自体も西和彦の名前がなくても有力な企業になっている。

話は長くなったが，その新聞記事によると，この西社長率いるアスキーが，転換社債という一種の借入金120億円の返済時期が迫っていることから，資金難で倒産の危機にあり，キヤノンが資本参加してこの苦境を救う話が進んでいる，との内容であった。

最終的にどうなるかは不明であるが，現時点では，この記事に沿った「最悪のシナリオ」は，キヤノンが降りたこともあって成立しなかった。その代わりにアスキーが多角化路線を撤退することで，取引銀行が150億円を追加融資して苦境を乗り切る筋書きとなっている。

アスキーがこの状況に陥ってしまったのは，いうまでもなく「バブル経済の崩壊」が原因である。まず来春に償還時期が迫っているという転換社債というのは，自社株に交換すれば購入者に返済しなくてもいい株式なのだが，なにしろ2万円を超えていたアスキーの株価がいまや1,000円以下なのだから，ほとんどの人が株に交換せずに換金をすると思われる。一連の株価暴落の影響だ。

さらにアスキーの負債や債務保証額が売上高を大きく上回る規模(合計すると500億円以上か)に達したのは，バブル絶頂時に土地を買い集めたり，海外の半導体やソフトの会社を買収したり，さらには直接コンピュータとは関係ないさまざまな事業に手を広げすぎたことが原因となっている。

つまり極論すると，株と土地という，まるでバブル崩壊の代名詞のような憂き目にあってしまったというわけである。もちろんこの背景には，西社長にブレーキをかける役割だった創業パートナーや有力社員が次々と退職してしまったという問題もあるのだが……。

説明が長くなったが，今回のアスキーの経営難問題は，いまになってみれば，アスキーもわずか100億円(あえて西社長の過去の活躍から「わずか」といいたい)でつまずく程度の中堅企業だったことを満天下に示してしまったことになる。

さて。

アスキーに限らない。

ハタと気がつくと，パソコンやコンピュータ関連で名を成してきた会社がこのところ，次々と倒産したり，経営難に陥っている。一時期はゲームソフトの大手だったハル研究所は先日倒産したばかりだし，どんぶり決算騒動で話題になったジャパンシステムやサイコムは経営難が続いている。CSKのような会社はいうにおよばず，NECや東芝，ソニー，パイオニア，沖電気工業などの超大手企業ですら相次ぎ減益や赤字になっているほどだから，よほどガッチリとした経営をしていないと，難しいご時勢だ。

だから，アスキーだけがどうこうということは決してないのだが，それでもシンボリックな企業であったアスキーがバブル崩壊の前につまずきを隠せなかったことは，少なくとも“「第1期パソコンドリーム」は1980年代いっぱいで終わっていた”，と追認させるに十分な出来事だった。実際，1990年代に入ってから，パソコン自体が以前ほどは話題にならなくなっている。小規模ソフトハウスのかげりは色濃くなっているし，身近なところではパソコン雑誌が次々と休刊になっている。

だが，ぼくはあくまでも「第1期」が終わっただけだと思うのだ。いまの状態はPC-9801を中心にしたマシンが「普通の電気製品」として普及し，ソフトもCDや図書のように地味に売れていく時期がしばらく続くのだろう。

しかしある時期に新しい側面から再びパソコンがスポットライトを浴びることは間違いない。それが1990年代後半なのか来世紀なのかはわからない。そのとき，最も素晴らしいハードなりソフトなりを提供できる会社や人が，「第2期パソコンドリーム」を手にするのだろう。

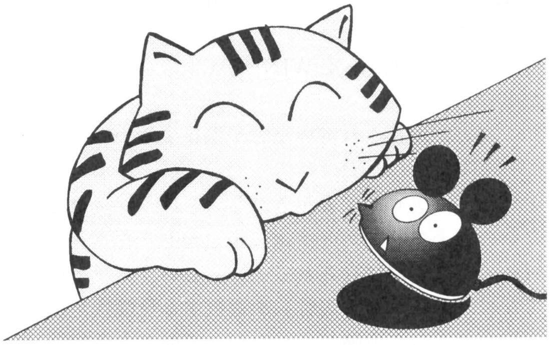
吾輩はX68000である

［第17回］

マウスで簡易アニメーション

Izumi Daisuke
泉　大介

吾輩と御仁を結ぶ
マウスカーソル
その妙技をお目にかけよう



吾輩は国産のパソコンとしてはめずらしくマウス標準装備で発売されたマシンである。以後今日まで，吾輩の携えたマウスはビジュアルシェル，グラフィックツール，アドベンチャーゲームやロールプレイングゲームなどで幅広く使われてきた。なかでも自慢なのは，マウスがアクションゲームにまで採用された点である。F14トムキャットが，諸兄の操作するマウスによって意のままに上昇・旋回する様子はひとかたならぬ感動をもたらしたことと思う。いったい，どこの誰がマウスで戦闘機を操縦できるなどと想像したことだろう。アフターバーナーは，スペースハリアーによって拓かれた高速(疑似)3Dスクロールゲームという新境地を完成の域に昇華させるとともに，マウスの斬新な使い方を示した特筆すべきゲームであった。うちの御仁もこれは大層お気に入りで，吾輩は連日連夜「ふぁ，ふぁ，ふぁ，ふぁいやー」とやらされたものである。

◆マウスとボタンとユーザーインタフェイス

吾輩はなにゆえマウス標準装備となったのか。それは，いまでは幻のシステムとなってしまったビジュアルシェルが，吾輩に標準で用意されていたからである。この機能強化版がOh!X付録ディスクのユーザーインタフェイスとして採用されているため，諸兄もお使いになったことがあるだろう。ビジュアルシェルは，シェルと呼ぶにはあまりにも機能が貧弱であった。実際，ビジュアルシェルにできることといえばファイル操作だけであり，スクロールバーまで用意されたウィンドウを使っていながら，その機能をユーザーが利用する道は閉ざされていたのである。

折しもキーボードとキャラクタ画面を利用するコンソール文化華やかなりし時代であり，環境変数を利用するUNIXやMS-DOS生まれのツールやアプリケーションが次々と移植されるという背景もあって，環境変数を操作することのできないビジュアルシェルはしだいに利用されなくなっていった。

そしてついに，本格的なウィンドウシステムであるSX-WINDOWにその座を譲った経緯は，諸兄もご存じのとおりである。

さて，吾輩に用意されたマウスはMS-DOSの世界で採用されている２ボタン式のマウスである。１ボタン式のマウスと比較すると，ボタンの数が増えた分だけより操作の自由度が増すというメリットがある反面，しっかりとしたガイドラインが設けられていないといたずらに操作が繁雑になるというデメリットも併せもつ。現状では左ボタンの機能はだいたい統制が取れているが，右ボタンの使い方は無政府状態と大差ない。結果，ユーザーは単純明快なユーザーインタフェイスを実現するはずのマウスの「使い方」を修得しなければならなくなってしまっている。

しかも，明確なルールなどのない，経験や勘が支配する「使い方」である。コマンドを実行するときに“／?”というオプションを指定すると使い方が表示されることがある。もしかすると“－?”か“／help”かもしれない。というのと同程度のものだ。こういったインタフェイスを脱却するためのGUIが，生まれながらにして泥沼に片足を突っ込んでいるというのは，なんとも皮肉な話ではないか。

白状すると，これは吾輩の意見ではない。なにを思ったのか，うちの御仁がある日突然Macintoshを携えて帰ってきた。後継機発表の噂が流れて安値を更新していたマシンをさらに買い叩いてきたのである。それ以来，マウスとユーザーインタフェイスの在り方について大層な好奇心を見せているのだ。

ご存じのようにMacintoshは１ボタンのマウスを採用している。御仁は，この制約がMacintoshの比類なきユーザーインタフェイスの原動力だというのである。なにせボタンがひとつしかないのだから，マウスは「選択」にしか使えない。そして「選択」しなければならないがゆえに，選択肢はすべて画面上に表示されていなければならない。ソフトを起動した瞬間に，そのソフトでなにができるのかは一目瞭然というわけである。右ボタンがソフトによってさまざまな使われ方をする，といった繁雑さが介入する余地はない。

画面に表示される選択肢でもMacintoshはユーザーへの思いやりをみせている。たとえばSX-WINDOWのエディタ.xでは，編集中の文書を保存しないままに終了しようとすると，

　　文書が保存されていないが本当に終了するか
　　［実行］［取消］
というダイアログが表示される。「これは親切だ。さすがはGUI」という感想をもたれた諸兄がひょっとしていらっしゃるかもしれないが，このダイアログは間違っている。なぜなら，将来SX-WINDOW用のワープロが登場したときに，
　　文書が保存されていないが保存するか
　　［実行］［取消］
というダイアログが表示されないという保証はまったくないのである。エディタ.xに慣れ親しんだユーザーは，冒頭の「文書が保存されて……」というメッセージを見てなにも考えずに「実行」を選択する危険性がある。逆の場合はさらに危険だ。したがって前者の選択肢は，
　　［終了］［取消］
であるべきだし，後者の選択肢は，
　　［保存］［取消］
であるべきなのである。

だいぶ前置きが長くなってしまった。これもSX-WINDOWを思えばこその老婆心である。好き嫌いは別として，GUIの先駆者たるMacintoshに見習うべきところは少なくない。マウスのボタンがひとつ増えたことで使い勝手のよさが2倍になるか，あるいはユーザーの負担が2倍になるか。それは今後のアプリケーション制作者にかかっているといっても過言ではない。下手をすると，「ユーザーの本能を逆撫でする」GUIになりかねないのだから。

◆マウスでアニメーション

ビジュアルシェルでウィンドウの順序を入れ換えたり，ウィンドウをクローズするとマウスカーソルが砂時計の形になるのをご存じであろう。単にマウスカーソルの形が変わるだけでなく，このとき，砂が落ちていくアニメーションが表示される。マウスを動かしてみればおわかりのように，マウスカーソルがアニメーションをしているのである。

マウスカーソルを書き換えるため余計な時間がかかるという難点はあるものの，いかにも「お待たせしています」という情緒が溢れていて，吾輩は気に入っている。なにか時間のかかる処理をしているときなどは，「こいつ，暴走してんじゃねーのか」という気遣いを多少なりとも防止する意味でも評価できよう。

図1　マウスカーソルのパターンを見る

```
-an 200000
00200000        moveq   #1,d1                 * 1番のパターンを選択
00200002        moveq   #$7b,d0
00200004        trap    #15
00200006        _exit
00200008        .
-g=200000                                     * プログラム実行
program terminated normally
-a 200000
  00200000      moveq   #$01,D1
                moveq   #2,d1                 * 2番のパターンを選択
  00200002      moveq   #$7B,D0
                .
-g=200000                                     * プログラム再実行
program terminated normally
-a 200000
  00200000      moveq   #$02,D1
                moveq   #3,d1                 * 3番のパターンを選択
  00200002      moveq   #$7B,D0
                .
-g=200000                                     * プログラム再実行
program terminated normally
-q
```

このアニメーションは，複数のマウスカーソルをとっかえひっかえ表示することで実現されている。どのようなパターンを表示しているのかお目にかけよう。まず，画面に“A>”などのプロンプトが表示されている状態で，マウスの右ボタンを押してみていただきたい。マウスカーソルが表示されるはずである。デバッガを起動してから，

```
moveq #1,d1
moveq #$7b,d0
trap  #15
_exit
```

というプログラムを入力して実行すると，マウスカーソルは砂時計に早変わりする。ここで使っているIOCSコール$7B_H$は，D1.Wで指定された番号のマウスカーソルを選択するという機能をもっている。マウスカーソルの0番はお馴染みの矢印型カーソル，そして1番にはこの砂時計が登録されているのである。

次にD1.Wに設定する数値を，2，3に変更して試してみていただきたい。よくよく注意していなければわかりづらいかもしれないが，砂が少しずつ落ちていくパターンが表示される。図1に作業の様子を示しておくので参照されたい。

では，砂が落ちていく1番から3番のマウスカーソルを使って，懸案のアニメーションを実行してみよう。これもIOCSコール$7C_H$にマウスアニメーション機能が用意されているので簡単に実現できる。IOCSコール$7C_H$は，

```
dc.w  1番目に表示するカーソル番号
dc.w  2番目に表示するカーソル番号
dc.w  3番目に表示するカーソル番号
  ⋮
dc.w  $FFFF
```

という形式でマウスのアニメーションパターンを用意し，この先頭アドレスをA1.Lにセットして呼び出すようになっている。これだけで，一定の時間ごとにマウスカーソルのパターンが置き換えられ，自動的にマウスがアニメーションを始めるようになるのである。実にお手軽だ。実際に試してみた例を図2に示しておく。短いプログラムなので，ぜひとも実験してみていただきたい。

図2のプログラムだが，この形式のリストは久しぶりなのでちょっと補足しておこう。諸兄がアセンブラなどを購入しなくても手軽に吾輩の機能を試していただけるように，吾輩はデバッガを使ったプログラムをお届けしているが，アセンブラをもっているという諸兄の便宜をも同時に図りたい。結果生まれたのがこのリスト形式なのである。デバッガで入力される諸兄は，“†”のついている行を無視して入力していただきたい。アセンブラをお使いの諸兄は“†”を取り去って入力すればOKである。もちろん，続く“.z0”が含まれている行を入力する必要はない。

プログラムを実行してみると，古参のユーザーの方にはお馴染みの砂時計が動き始める。パターンの変化が早

すぎてよく見えないという方は，

```
dc.w  1
dc.w  1
dc.w  2
dc.w  2
dc.w  3
dc.w  3
   ⋮
```

というようにパターンを２つずつ，あるいはもうひとつ増やして３つずつ並べられるとよかろう。砂が落ちていく様子をじっくり観察できるはずである。

◆アニメーションを自作する

Macintoshのソフトウェアでは，CPUが時間のかかる処理を実行する際には大抵なんらかのアニメーションが表示されるようになっている。棒グラフが100％に向かって伸びていくものもあれば，吾輩の砂時計のようにマウスカーソルがアニメーションを行うものもある。「コンピュータは動いています」とユーザーに通知することは，彼のマシンでは基本作法のひとつなのである。マウスカーソルのアニメーションの中には，思わず笑ってしまうような秀作も少なくない。とりわけ，シェアウエアやフリーウェアには気のきいたものが多いようである。ファイルを読み込んで展開している間，人が走り続けているものは，御仁のお気に入りのひとつとなっている。ファイルの読み込みを一時停止したときには，この人型が正面を向いて左足をイライラと足踏みする。その姿が可愛いというのである。

標準で用意されているマウスカーソルのアニメーション機能なれば，これで遊ばないのは宝のもち腐れというものだ。マウスカーソルはIOCSコール$7A_H$を使えば自分で好きなように設定できるようになっている。カーソルのデータを用意し，その先頭アドレスをA1.Lに，そして，そのデータをセットするマウスカーソルの番号をD1.Wに入れて呼び出すだけだ。なお，マウスカーソルのデータは以下のようになっている。

```
dc.w  Xオフセット,Yオフセット
dc.w  白色のパターン1
dc.w  白色のパターン2
dc.w  白色のパターン3
         ⋮
dc.w  白色のパターン16
dc.w  青色のパターン1
dc.w  青色のパターン2
dc.w  青色のパターン3
         ⋮
dc.w 青色のパターン16
```

最初のオフセットというのは，マウスの座標を得る場合に16×16ドットのパターンのどのドットをマウスの座標とするか，という設定である。たとえばパターンの中心をマウスの座標としたいのなら，(7,7)か(8,8)にしておけばいい。

続いてマウスカーソルのベースの色である白のパターンである。これは１ドットを１ビットで表現するようになっているので，白の点線にしたいなら，

```
dc.w  _0101010101010101
```

とすればいい。これは1ライン分のデータなので，あと15回繰り返せば白のパターンが完成することになる。ここで注意しておきたいのは，データが０のところが白のドットになり，データが１のところは透明色となることだ。なお，データの先頭についている“_”は，デバッガで２進数を指定する記号である。アセンブラでは“%”を使うようになっているので注意されたい。

白のパターンが16ライン分続いたあとは青のパターンである。パレットの設定によってはマウスカーソルが茶色で表示されるかもしれないが，その場合は茶色のパターンということになる。白のパターンと同様に指定するが，こちらはデータが１になっているところが青のドットになる。

さて，マウスカーソルの作成法だが，まず紙に16×16のマス目を書いていただきたい。この上で，青は○，白は×というようにパターンを作成し，それをdc.wのデータ列に直せば出来上がりである。こうして作ったデータを実際にマウスカーソルとしてセットしているのが図３のリストである。

とりあえず，データは基本の１パターンのみ用意しておいた。あとは回転するパターンをいくつか作って登録しておけば，IOCSコール$7C_H$を使って簡単に回転マウスカーソルを作成することができる。諸兄の創意と工夫で面白いマウスアニメーションを作成してみていただきたいものだ。

と，突き放してしまうのも気が引ける。紙の上でパターンを作成するのはいいとしても，そのあとでdc.wのデータ列に直すのはあまりに面倒だと思われることだろう。こんなとき御仁はX-BASICを使って簡単なツールを作ってしまう。紙にパターンを作成し手作業でdc.wのデータ列に直す10倍の時間がプログラム作成にかかるとしても，である。

今回のようなアニメーションするために，いくつかのパターンを作成しなければならないという局面にいたって，御仁は紙の上でさっさとやってしまったほうが短時間でできるなどということを顧みもしない。まったく，このあたりの価値観にはいつもながら呆れ果てる。いわば，「楽をするためにはどんな努力でもする。楽をするための苦労は厭わない」というわけである。

図2　砂時計のアニメーション

```
-z0=200000
-an .z0
        †_exit              equ      $ff00

00200000          moveq    #$70,d0            * マウス初期化
00200002          trap     #15
00200004          moveq    #$71,d0            * マウスカーソル表示
00200006          trap     #15
        †         lea      mspat,a1           * アニメーションパターン設定
00200008          lea      .z0+$14,a1
0020000E          moveq    #$7c,d0            * アニメーション開始
00200010          trap     #15
        †         dc.w     _exit
00200012          _exit
        †mspat:
00200014          dc.w     1
00200016          dc.w     2
00200018          dc.w     3
0020001A          dc.w     $ffff
0020001C          .
```

図3　自作のマウスカーソルを表示する

```
-z0=200000
-an .z0
        †_exit              equ       $ff00

00200000        moveq    #$70,d0              * マウス初期化
00200002        trap     #15
00200004        moveq    #$71,d0              * マウスカーソル表示
00200006        trap     #15
00200008        moveq    #1,d1                * カーソル番号
        †       lea      msdata,a1           * マウスカーソルデータ
0020000A        lea      .z0+$1c,a1
00200010        moveq    #$7a,d0              * パターンセット
00200012        trap     #15
00200014        moveq    #1,d1                * 選択するカーソル番号
00200016        moveq    #$7b,d0              * マウスカーソル選択
00200018        trap     #15
        †       dc.w     _exit
0020001A         _exit
        †msdata:
0020001C        dc.w     7,7
00200020        dc.w     _1111110001111111
00200022        dc.w     _1111001110011111
00200024        dc.w     _1110111111101111
00200026        dc.w     _1101111111110111
00200028        dc.w     _1011111111111011
0020002A        dc.w     _1011111011111011
0020002C        dc.w     _0111111011111101
0020002E        dc.w     _0111100000111101
00200030        dc.w     _0111111011111101
00200032        dc.w     _1011111011111011
00200034        dc.w     _1011111111111011
00200036        dc.w     _1101111111110111
00200038        dc.w     _1110111111101111
0020003A        dc.w     _1111001110011111
0020003C        dc.w     _1111110001111111
0020003E        dc.w     _1111111111111111
00200040        dc.w     _0000000000000000
00200042        dc.w     _0000001110000000
00200044        dc.w     _0000111111100000
00200046        dc.w     _0001111111110000
00200048        dc.w     _0011111111111000
0020004A        dc.w     _0011111011111000
0020004C        dc.w     _0111111011111100
0020004E        dc.w     _0111100000111100
00200050        dc.w     _0111111011111100
00200052        dc.w     _0011111011111000
00200054        dc.w     _0011111111111000
00200056        dc.w     _0001111111110000
00200058        dc.w     _0000111111100000
0020005A        dc.w     _0000001110000000
0020005C        dc.w     _0000000000000000
0020005E        dc.w     _0000000000000000
00200060
```

もっとも，このような性向があればこそ，不精な御仁でも人並みの生活が送れているのだろう。そして吾輩は，御仁のプログラミングを楽しむことができるというわけである。

紹介するマウスカーソルの作成プログラム(リスト1)は，御仁が簡易表集計ソフトを作った折に用意したものである。ユーザーインタフェイス関係に話が及んだ手前，少々手直しをしてある。使い方は簡単で，右側のパレットから色を選び左側のパターン作成用紙に貼り付けていくだけである。使用するマウスのボタンは左だけ。パターンが作成できたら“save”を選ぶと，アセンブラ用のdc.wデータファイルを作成するようになっている。デバッガでご利用の諸兄は，エディタでこのファイルを読み込んだあと，

```
z0=200000
an .z0
moveq #$70,d0
trap #15
   ⋮
```

のようにデバッガに入力する命令をファイルの先頭に付け加え，デバッガの“<”コマンド用のファイルを作成して試していただきたい。

もちろんデータの先頭の“%”を，2進数を意味する“_”に変更するのもお忘れなく。楽しいアニメーションができたら，ぜひとも吾輩に教えていただきたい。御仁とともに楽しみたいと思っている。

リスト1　簡易マウスパターンエディタ

```
 10 int i
 20 int pencolor
 30 int xpos,ypos,lbtn,rbtn
 40 char ptn(16,16)
 50 str filename
 60 /*
 70 screen 2,0,1,1
 80 mouse(0)
 90 msarea(0,0,767,511)
100 /*
110 /*      画面作成
120 /*
130 for i=0 to 16
140   line(i*20,0,i*20,319,15)
150   line(0,i*20,319,i*20,15)
160 next
170 box(400,0,448,20,15)
180 box(400,30,448,50,15)
190 fill(401,31,447,49,3)
200 fill(400,60,448,80,15)
210 locate 51,17
220 print "save"
230 box(400,272,448,290,15)
240 locate 51,19
250 print "exit"
260 box(400,304,448,322,15)
270 mouse(1)
280 /*
290 /*      ボタン判定
300 /*
310 while 1
320   msstat(xpos,ypos,lbtn,rbtn)
330   if ( lbtn ) then {
340     mspos(xpos,ypos)
350     if ( xpos < 320 ) then {
360       setdot(xpos,ypos)
370     } else if ( ypos < 322 ) then {
380       if ( selectfnc(xpos,ypos) ) then {
390         break
400       }
410     }
420   }
430 endwhile
440 end
450 /*
460 /*      ドットを描く
470 /*
480 func setdot(x,y)
490   if ( y < 320 ) then {
500     x = x / 20
510     y = y / 20
520     ptn(x,y)=pencolor
530     fill(x*20+1,y*20+1,x*20+19,y*20+19,pencolor)
540   }
550 endfunc
560 /*
570 /*      save/exitの選択
580 /*
590 func selectfnc(x,y)
600   if ( x<400 or 448<x ) then return( 0 )
610   if ( 0<y and y<20 ) then {
620     pencolor=0 : return( 0 )
630   }
640   if ( 30<y and y<50 ) then {
650     pencolor=3 : return( 0 )
660   }
670   if ( 60<y and y<80 ) then {
680     pencolor=15 : return( 0 )
690   }
700   if ( 272<y and y<290 ) then {
710     savefile() : return( 0 )
720   }
730   if ( 304<y and y<322 ) then {
740     return( 1 )
750   }
760 endfunc
770 /*
780 /*      ファイル出力
790 /*
800 func savefile()
810   int i,j,k,check
820   int bdata(16)
830   int fp
840   locate 0,22
850   input "filename : ",filename
860   fp=fopen(filename,"c")
870   locate 0,22
880   print chr$(5);
890   for j=0 to 15
900     print ".";
910     fwrites(chr$(9)+"dc.w"+chr$(9)+"%",fp)
920     for i=0 to 15
930       if ( ptn(i,j)=15 ) then {
940         fwrites("0",fp)
950       } else {
960         fwrites("1",fp)
970       }
980     next
990     fwrites(chr$(13)+chr$(10),fp)
1000   next
1010   for j=0 to 15
1020     print ".";
1030     fwrites(chr$(9)+"dc.w"+chr$(9)+"%",fp)
1040     for i=0 to 15
1050       if ( ptn(i,j)=3 ) then {
1060         fwrites("1",fp)
1070       } else {
1080         fwrites("0",fp)
1090       }
1100     next
1110     fwrites(chr$(13)+chr$(10),fp)
1120   next
1130   fclose(fp)
1140   locate 0,22
1150   print chr$(5)
1160 endfunc
```

大きな数の話

Murata Toshiyuki 村田 敏幸

今回は，先月解説した整数演算の話を多倍長データに拡張させていきます。感覚ではつかみ取れない数を扱うことでとまどうかもしれませんが，基本は変わりません。また，多倍長整数演算の応用として，固定小数点でπの値を求めるプログラムも紹介します。

今回は巨大な数を扱うことを考える。いわゆる多倍長整数の四則演算ルーチンを作成してみよう。多倍長整数演算はそれほどよく使われる処理というわけではないが，プログラムの題材としてはいろいろ考えさせる点もあり，なかなか面白い。テーマ自体に興味のもてない読者も，純粋なパズルのつもりで脳みそを絞ってみてもらいたいと思う。

なお，今回は話の都合で0および正の値のみを扱う。負数への対応は比較的容易だから，余裕のある読者は検討してみるとよいだろう。

多倍長整数を扱うために

さっそくリスト1に今回採用した多倍長整数のデータ構造定義を示す。数値データ自体はふつう整数を自然に拡張した純2進表現でモトローラ式の語順，つまり，アドレスの若いほうに高位桁を置く形式[1]で並べる。この先頭に，数値の上位桁の0を省いた正味桁数を覚えておく1ワードのフィールドが設けてある。上限ぎりぎりではない比較的小さな値を使用する際に上位桁を0で埋める手間や，この余分な0まで処理する手間を省こうという狙いだ。なお，先頭フィールドに格納するのは正味データが何ロングワードあるかという値，いい換えると2^{32}進法での桁数とする。上位桁の0をすべて省くので，0という値は桁数0として表現される。

桁数の最大値は3行の記号定数BNMAXで示される。BNMAXもロングワード単位の値で指定する。実際の最大桁数は2進でBNMAX×32桁，10進だとその$\log_{10}2$倍のBNMAX×約9.633桁となる。リスト1では仮にBNMAXを106（10進1000桁＋若干のゆとり分）に定義してあるが，用途・目的に応じて1～32767の範囲で再定義してかまわない。32767にすれば10進で30万桁の数が表現できる。なお，上限が32767になっているのは，一部の演算ルーチンがデータをワード単位で処理する都合だ。ワード単位の桁数はロングワード単位の桁数の2倍になるわけだが，BNMAXが32767以下なら2倍しても16ビットで収まり，ループカウンタとする際にdbraが利用できる[2]。

ところで，このBNMAXのように書き換えて使う記号定数は，何かのはずみで変な値に再定義されてしまう可能性がある。そういった事故が起きないよう，リスト1では7行の.fail疑似命令でBNMAXの範囲を制限している。.failは故意にエラーを起こすAS.Xの疑似命令で，後ろに書いた条件式が成り立つとアセンブルが中止される。リスト1ではBNMAXが1より小さいか32767より大きいときにエラーが起きるようになっている。この行はBNMAXの最小値と最大値を示すコメントとしての役割も果たしている，といえるだろう。

.fail疑似命令の使用例をもうひとつ示しておこう。リスト2はスタックフレームの定義例だが，.offsetの基準値に間違いがある。このままでは引数が正しく取得できないばかりか，linkで待避したa6レジスタの内容が破壊され，サブルーチンからのリターン時にバスエラーかアドレスエラーが発生することになるだろう。この種のバグはソースを見ているだけではなかなか発見できないが，15行のような.fail疑似命令を入れておけば，アセンブル時に誤りに気づくというわけだ。

さて，データ構造を決めたところで，ユーティリティ的なサブルーチンをひとつ作成しておく。リスト3に示すbncopyは，リスト1で定義した形式の多倍長整数変数を別の領域にコピーするサブルーチンだ。

pea.l　転送元先頭アドレス
pea.l　転送先先頭アドレス

1）　多バイト数値データのメモリ上での並び順を時にエンディアン（endian）と呼び，いわゆるモトローラ式をリトルエンディアン（little endian），インテル式の若いアドレスに下位バイトを置く方式をビッグエンディアン（big endian）という。

2）　今回の演算ルーチンを負数対応にする際には，この桁数を格納するフィールドの余った1ビットを符号の保持用に利用できるだろう。

リスト1　BNDEF.H

```
 1: *           多倍長整数演算のデータ構造定義
 2:
 3: BNMAX   equ     106                     *最大桁数（2^32進）
 4:                                         *実際の桁数は
 5:                                         *  2進でBNMAX×32桁
 6:                                         *  10進でBNMAX×9.633桁
 7: .fail   (BNMAX.lt.1).or.(BNMAX.gt.32767)
 8: *
 9:         .offset 0
10: *
11: BNWORD: .ds.w   1                       *実際に使用している桁数
12: BNVAL:  .ds.l   BNMAX                   *純2進値
13: BNVAR:                                  *多倍長整数変数記憶領域サイズ
14: *
15:         .text
```

```
        jsr     bncopy
        addq.l  #8, sp
```
のようにして使う。

プログラムの上では特に問題になる点はないだろう。まず，20～21行で転送元と転送先が一致している場合を弾く。続いて23～25行で桁数を示す１ワードをコピーする。この時点で桁数が０だったらサブルーチンを抜ける。そうでなければ，27～32行でデータ部の最上位桁位置を求める。BNMAX桁のうちn桁だけを使用している場合，上位BNMAX－n桁はコピーする必要がないからこの部分を飛ばすわけだ。あとは求めた位置から35～36行のループ中のmoveで，桁数分のデータ転送をしている。

加算

定石どおり加算ルーチンから始めよう。アルゴリズムとしては，データを下位から適当な単位で分割し，それぞれを１桁の数だと思って通常の筆算と同じ感覚で下位桁から順に足していけばよい。68000の32ビット加算命令を生かすことを考えると，分割の単位も32ビットにするのが自然だろう。

多倍長整数の加算では，繰り上がりの扱いがポイントだ。各桁どうしを加える際には，ひとつ下の桁どうしを足したときの繰り上がりを一緒に足し込む必要がある。ふつうにやろうとすると，各桁をadd命令で足したときのＣビットにより処理を振り分けて，C＝1のときには次の桁へ１を余分に足す，というかたちで実現することになるだろう。しかし，68000には多倍長整数どうしの加算を想定した繰り上がり込みの加算命令addxがあり，同様の操作をずっと簡単に行うことができる。

addxは，加算時にccrのＸビットを一緒に足し込む命令だ。Ｘビットは加減算やシフト時に繰り上がりや桁借りがあると１になる[3]。したがって，最下位桁のみをaddで足し，２桁目以降はaddの代わりにaddxを使えば，直前の桁を加えたときの繰り上がりを同時に足し込むことができる。ただし，addxに適用できるアドレッシングモードは非常に限られていて，

```
        addx  dn,dn
```
のようなデータレジスタどうしか，
```
        addx  -(an),-(an)
```
のようなプリデクリメント付きのアドレスレジスタ間接形式によるメモリどうしの形式しかサポートされていない。

では，リスト４を見てもらおう。このサブルーチンbnaddは，
```
        pea.l   加数
        pea.l   被加数
        pea.l   結果格納先
        jsr     bnadd
        lea.l   12(sp), sp
```
のようにして呼び出す。結果格納先は被加数や加数と同一の領域を指していてもかまわないので，
```
        pea.l   A
        pea.l   A
        pea.l   A
        jsr     bnadd
        lea.l   12(sp), sp
```
とすれば，

A＝A＋A

相当の処理も行える。

リスト４ではあとの処理の都合で，真っ先に被加数と加数との桁数を比べ，被加数のほうが桁数が大きくなるよう必要に応じて両者を交換している

リスト2　FAILTEST.S

```
*           .failの使用例

            .offset -10
*
WORKSIZ:
WORK1:  .ds.w   1       *
WORK2:  .ds.w   1       *
WORK3:  .ds.l   1       *
WORK4:  .ds.l   1       *
_A6:    .ds.l   1       *a6待避用
_PC:    .ds.l   1       *pc待避用
ARG1:   .ds.l   1       *引数1
ARG2:   .ds.w   1       *引数2
*
.fail   _A6.ne.0
*

        .text
        .even
*
aproc:
        link    a6,#WORKSIZ

        movea.l ARG1(a6),a0
        move.w  ARG2(a6),d0

*               :

        unlk    a6
        rts

        .end
```

リスト3　BNCOPY.S

```
 1: *           多倍長整数の複写
 2:
 3:             .include        bndef.h
 4: *
 5:             .xdef   bncopy
 6: *
 7:             .offset 4
 8: *
 9: DEST:       .ds.l   1               *転送先
10: SOUR:       .ds.l   1               *転送元
11: *
12:             .text
13:             .even
14: *
15: bncopy:
16:             movem.l d0-d1/a0-a1,-(sp)
17: SAVSIZ      =       (2+2)*4
18:             movem.l DEST+SAVSIZ(sp),a0-a1
19:
20:             cmpa.l  a0,a1           *転送先 = 転送元なら
21:             beq     retn            *  すぐ抜ける
22:
23:             move.w  (a1)+,d0        *値が0なら
24:             move.w  d0,(a0)+        *
25:             beq     retn            *  コピー完了
26:
27:             moveq.l #0,d1           *最上位桁位置を求める
28:             move.w  #BNMAX,d1       *
29:             sub.w   d0,d1           *
30:             lsl.l   #2,d1           *
31:             adda.l  d1,a0           *
32:             adda.l  d1,a1           *
33:
34:             subq.w  #1,d0           *桁数分転送する
35: loop:       move.l  (a1)+,(a0)+     *
36:             dbra    d0,loop         *
37:
38: retn:       movem.l (sp)+,d0-d1/a0-a1
39:             rts
40:
41:             .end
```

3)　これはＣビットの変化と同じだが，Ｃビットはmove命令などでも変化するのに対して，Ｘビットは加算命令，減算命令，符号反転命令，シフト命令，Ｘビット込みのローテート命令以外では変化しない。Ｃビットは繰り上がり・桁借りがあったかどうかを示す純粋なフラグで，Ｘビットは繰り上がり・桁借りの短期間の保存目的で存在すると考えてよい。

4)　効率の点で最良ではない。
```
        addx.l  -(an),-(an)
```
を利用するのは被加数と結果格納先が最初から一致している場合に限り，そうでなければあらかじめ被加数を結果格納先にコピーしたりせずに，move命令で１桁取り出しては，
```
        addx.l  dn,dn
```
で足して結果格納先に転送するようにしたほうが多少速い。

(22～27行)。加算では，

A＋B＝B＋A

という交換則が成り立つのはいうまでもない。そののち，29～33行で被加数を結果格納先にコピーする。これは，

addx.l －(an), －(an)

を有効に使うための処置だ[4]。被加数と結果格納先が最初から一致している場合，このコピーは不要だが，bnadd側では余計なことは考えず，先に用意したbncopyにコピー処理を（する必要があるかどうかを含めて）一任している。

35行で被加数と加数の桁数の差を求めている理由についてはあとで触れる。36～37行で加数の桁数を保持するd2から１を引いてdbraの使用に備え，同時に加数が０の場合を弾いている。続く40～42行で被加数，加数の最下位桁位置を求める。プリデクリメントする都合上，求めたアドレスは最下位桁位置の直後のメモリを指している。ここで先頭アドレスに加えているBNSIZEはリスト１のBNDEF.Hで定義された多倍長整数変数の総バイト数だ。

ここまでで前処理が終わって，44～46行がメインループ。ループ中ではaddxを使って下位桁から順に加数の桁数分の繰り上がりを考慮した加算を行っている。さかのぼった36行でこっそりccrのXビットをクリアしている[5]ので，繰り上がりを足す必要のない最下位桁を特別扱いせずにすんでいることに注目してほしい。

44～46行のループを抜けたら，49行以降で加数の最上位桁を加えたときに発生するかもしれない繰り上がりを処理する。被加数の上位ビットに１が詰まっていると連鎖的に繰り上がりが発生することに注意し，49～51行では最大で「被加数の桁数－加数の桁数」に相当する桁数分の繰り上がりが正しく処理できるよう，先に35行で求めておいた桁数の差をループカウンタにしてdbccによるループを構成している。なお，47行でループ末尾のdbccの位置に飛び込んでいるのには数重の意味があるのでよくかみしめてもらいたい。

２つ目のループも抜けて51行に達した時点で，加算の処理はほとんど終わっている。残るは被加数の最上位桁からの繰り上がりだ。この繰り上がりがあった場合，加算結果は被加数よりも１桁大きくなるので，オーバーフローに対して配慮する。被加数がBNMAX桁だったらこれ以上の上位桁を格納するスペースがないわけだから，54～55行で弾いて，ccrのNビットを立ててエラーを返す（という仕様）。まだ桁数のゆとりがあったら，56～58行で最上位桁に１を入れて，桁数を１増やす。

減算

加算ルーチンができたところで，すかさず減算ルーチンに取り掛かる。考え方は加算の場合とまったく同じ。下位桁から１桁ずつ桁借りともども引いていくという筆算スタイルだ。桁借りを含む減算にはaddxと対になるsubx命令が利用できる。

で，リスト５。大筋はリスト４と変わらないので，減算特有の点についてのみ触れよう。まず，28～32行。今回は負数を扱わないので，被減数と減数の桁数を比べて，

被減数の桁数＜減数の桁数

だった場合はさっさとエラーを返している。また，被減数から減数を引き終わり，桁借りの処理も終わった49行の時点でなおかつ上位桁からの借り入れがあったら，被減数＜減数なのでやはりエラーを返す。あと，51～58行では減算結果の正味桁数を求めている。いま採用しているデータ構造に従って，求めた差の上位桁の０を省くわけだ。

乗算

今度は乗算だ。乗算もまた，多倍長整数を適当な

5) 「こっそり」やらずに，ループの直前にＸビットをクリアするだけでほかに影響のない，

subi.w #0, d0

のような命令か，あるいは，ccrを直接操作して全フラグをリセットする，

move.w #0, ccr

を置いたほうがわかりやすいプログラムにはなる。が，リスト４では効率優先で余計な命令は置かず，代わりにコメントを入れて逃げている。

リスト4 BNADD.S

```
 1: *           多倍長整数+多倍長整数
 2:
 3:             .include        bndef.h
 4: *
 5:             .xdef   bnadd
 6:             .xref   bncopy
 7: *
 8:             .offset 4
 9: *
10: RESULT:     .ds.l   1           *和
11: A:          .ds.l   1           *被加数
12: B:          .ds.l   1           *加数
13: *
14:             .text
15:             .even
16: *
17: bnadd:
18:             movem.l d0-d2/a0-a2,-(sp)
19: SAVSIZ      =       (3+3)*4
20:             movem.l RESULT+SAVSIZ(sp),a0-a2
21:
22:             move.w  (a1),d1             *被加数の桁数≧加数の桁数
23:             move.w  (a2),d2             *  を保証する
24:             cmp.w   d2,d1               *
25:             bcc     skip                *
26:             exg.l   a1,a2               *
27:             exg.l   d1,d2               *
28:
29: skip:       pea.l   (a1)                *被加数を結果格納領域に
30:             pea.l   (a0)                *  コピーする
31:             jsr     bncopy              *
32:             addq.l  #8,sp               *
33:             movea.l a0,a1               *
34:
35:             sub.w   d2,d1               *d1 = 被加数の桁数-加数の桁数
36:             subq.w  #1,d2               *dbraを考慮
37:             bcs     done                *加数が0なら終了
38:                                         *C = X = 0
39:
40:             move.l  #BNVAR,d0           *最下位桁位置を求める
41:             adda.l  d0,a1               *
42:             adda.l  d0,a2               *
43:
44: loop1:      addx.l  -(a2),-(a1)         *繰り上がり込みで
45:                                         *  最下位桁から加えていく
46:             dbra    d2,loop1            *加数の桁数分繰り返す
47:             bra     next2
48:
49: loop2:      addq.l  #1,-(a1)            *繰り上がりを処理する
50: next2:      dbcc    d1,loop2            *
51:             bcc     done                *最終的な繰り上がりが
52:                                         *  なければ終了
53:
54:             cmpi.w  #BNMAX,(a0)         *オーバーフロー?
55:             bcc     flow                *
56:             moveq.l #1,d0               *最終的な繰り上がり
57:             move.l  d0,-(a1)            *
58:             addq.w  #1,(a0)             *桁数++
59:
60: done:       moveq.l #0,d0               *N = 0
61: retn:       movem.l (sp)+,d0-d2/a0-a2
62:             rts
63:
64: flow:       moveq.l #-1,d0              *N = 1
65:             bra     retn
66:
67:             .end
```

リスト5 BNSUB.S

```
 1: *        多倍長整数－多倍長整数
 2:
 3:          .include       bndef.h
 4: *
 5:          .xdef   bnsub
 6:          .xref   bncopy
 7: *
 8:          .offset 4
 9: *
10: RESULT:  .ds.l   1          *差
11: A:       .ds.l   1          *被減数
12: B:       .ds.l   1          *減数
13: *
14:          .text
15:          .even
16: *
17: bnsub:
18:          movem.l d0-d2/a0-a2,-(sp)
19: SAVSIZ   =       (3+3)*4
20:          movem.l RESULT+SAVSIZ(sp),a0-a2
21:
22:          pea.l   (a1)               *被減数を結果格納領域に
23:          pea.l   (a0)               *  コピーする
24:          jsr     bncopy             *
25:          addq.l  #8,sp              *
26:          movea.l a0,a1              *
27:
28:          move.w  (a1),d1            *被減数の桁数≧減数の桁数
29:          move.w  (a2),d2            *  を保証する
30:          beq     retn               *減数が0なら終了
31:          cmp.w   d2,d1              *
32:          bcs     flow               *
33:
34:          move.l  #BNVAR,d0          *最下位桁位置を求める
35:          adda.l  d0,a1              *
36:          adda.l  d0,a2              *
37:
38:          move.w  d1,d0              *d0 = 被減数の桁数
39:          sub.w   d2,d1              *d1 = 被減数の桁数-減数の桁数
40:          subq.w  #1,d2              *dbraを考慮
41:                                     *C = X = 0
42: loop1:   subx.l  -(a2),-(a1)        *桁借り込みで
43:                                     *  最下位桁から引いていく
44:          dbra    d2,loop1           *減数の桁数分繰り返す
45:          bra     next2
46:
47: loop2:   subq.l  #1,-(a1)           *桁借りを処理する
48: next2:   dbcc    d1,loop2           *
49:          bcs     flow               *被減数<減数だった
50:
51:          sub.w   d1,d0              *上位桁の0を省く
52:          subq.w  #1,d0              *
53: loop3:   tst.l   (a1)+              *
54:          dbne    d0,loop3           *
55:          addq.w  #1,d0              *
56:
57:          add.w   d1,d0              *差の桁数を求めて
58:          move.w  d0,(a0)            *  格納
59:                                     *N = 0
60:
61: retn:    movem.l (sp)+,d0-d2/a0-a2
62:          rts
63:
64: flow:    moveq.l #-1,d0             *N = 1
65:          bra     retn
66:
67:          .end
```

リスト6 BNMUL_W.S

```
 1: *        多倍長整数×16ビット整数
 2:
 3:          .include       bndef.h
 4: *
 5:          .xdef   bnmul_w
 6: *
 7:          .offset 4
 8: *
 9: RESULT:  .ds.l   1          *積
10: A:       .ds.l   1          *被乗数
11: B:       .ds.w   1          *乗数
12: *
13:          .text
14:          .even
15: *
16: bnmul_w:
17:          movem.l d0-d3/a0-a1,-(sp)
18: SAVSIZ   =       (4+2)*4
19:          movem.l RESULT+SAVSIZ(sp),a0-a1
20:
21:          move.w  (a1),d1            *被乗数が0なら積は0
22:          beq     zero               *
23:          move.w  B+SAVSIZ(sp),d2    *乗数が0なら積は0
24:          beq     zero               *
25:
26:          move.w  d1,(a0)            *積の桁数を格納
27:
28:          move.l  #BNVAR,d0          *最下位桁位置を求める
29:          adda.l  d0,a0              *
```

単位で区切り，各単位を１桁の数とみなして筆算を真似るという方針で解決できる。ここで，68000では16ビット数どうしの乗算命令しかサポートされていないので，分割の単位は選択の余地なく16ビットになる。

多倍長整数どうしの乗算ルーチンを示す前に，前段階としてリスト６を見てもらおう。リスト６のサブルーチンbnmul_wは多倍長整数に16ビット整数を掛ける。2^{16}進で考えれば，多桁×１桁の乗算ルーチンだ。35行までの前処理部分は見てのとおりだからいいとして，被乗数の最下位桁から１桁取り出しては乗数を掛け，積を１桁ずつ求める36～42行のメインループはじっくり読んでもらいたい。ここでは，「大きな繰り上がり」と「小さな繰り上がり」ともいうべき２種類の繰り上がりを正しく扱うために十分気を遣っている。１桁×１桁を計算したら（36～37行），得られる２桁の積の上位桁(＝大きな繰り上がり)をd3レジスタに覚えておき（40～41行），次の桁の乗算結果に38行で足す。この加算時に発生した繰り上がり（＝小さな繰り上がり）はXビットに保持され，ループ１周後の38行で足し込まれる。ループ中には38行以外にXビットを変化させる命令がないことを確認しておいてほしい。

なお，ｎ桁×１桁の積はｎ桁かn＋1桁になる。36～42行のループではこのうち下位ｎ桁のみを求めた形になっており，n＋1桁目は44～51行で処理する。メインループ中で最後に発生した「小さな繰り上がり」を取り込むための45行に注意してもらえれば，何をやっているかは容易に読み取れるだろう。

こうして，多桁×１桁の乗算ができるようになれば，多桁×多桁の乗算ルーチンまであとわずかだ。多桁×多桁の乗算は，被乗数に乗数の各１桁を掛け，それぞれの部分積を桁位置を揃えて足すことで実現される。実例としてサブルーチンbnmulをリスト７に示そう。74～86行が被乗数×乗数１桁を求める処理で，これを72～89行のループで括った格好になっている。被乗数×乗数１桁を個別に求めてあとから足すのではなく，結果格納先へ次々と足し込むようにした関係で，78～85行のループはリスト６よりも若干複雑になっているが，やっていることは変わらない。また，同じ理由で，結果格納先はあらかじめゼロクリアしておかなければならなくなっている（57～61行）。

メインループ中の小さなポイントとして，88行を挙げておこう。部分積を足し込む結果格納先の桁位置を１桁分ずらすことで，部分積の重みに応じて桁位置を揃える処理を実現している。

さて，リスト７ではメインループ以外の細々とした部分にも目を向けてもらいたい。まず，24～31行では通常の筆算のときと同様，被乗数のほうが乗数よりも桁数が大きくなるよう，必要に応じて両者を交換している。あまり深く考えずに「筆算ではこうするから」というだけの理由で付け加えた処理だが，たぶん，効率の点でもいくらかよい結果が得られる

だろう。また，33～49行では結果の格納先と被乗数や乗数が同一のメモリを指していたときに，上書きを避けるため，被乗数や乗数を作業用のメモリにコピーしている。本質的な処理ではないが，こうしないと，

```
	pea.l	A
	pea.l	A
	pea.l	A
	jsr	bnmul
	lea.l	12(sp),sp
```

のような呼び出しができなくなり，すでに作成した演算ルーチン群との対称性が崩れてしまう。なお，コピー先の作業用領域は別ファイルのリスト8，9に抜き出してある。

オーバーフローの判定周りでは少々強引なこともやっている。n桁×m桁の積はn＋m桁か，n＋m－1桁になることから，明らかにオーバーフローする場合を演算開始前の51～55行で弾くのはよい。しかし，これだけでは判断できず，オーバーフローするかどうかが微妙なときはそのまま積を求める処理になだれ込んでいる。最悪の場合，得られる積はBNMAXで決められた桁数の上限よりも大きくなり，その結果格納先のデータ部から1ワードはみ出す可能性がある。もっとも，はみ出したとしても，多倍長整数変数にはデータ部の直前に桁数を格納するフィールドがあり，ここが緩衝地帯として働く。つまり，演算に先立って桁数を格納するフィールドを0にしておき(58行)，積を求めたあとで桁数格納フィールドに積がはみ出したかどうかを調べ(92～93行)，はみ出していなかったら別に保持していた桁数を収める(96～101行）というごまかしがきくわけだ。

除算

除算についても，本格的な多倍長整数どうしの除算ルーチンを示す前に多倍長整数÷16ビット整数の除算ルーチンbndiv_wをリスト10に用意してみた。余りはd0に返る。

リスト10では多倍長整数を 2^{16}進数とみなして，上位桁から順に割り，商を1桁ずつ立てるという構造になっている。44～47行のループ中では，通常の筆算同様，直前の桁を処理した時点での余りと被除数の1桁を合わせて（1桁「降ろしてきて」）2桁の数を作っては除数で割っている。あとは，商の桁数の調整をして（52～58行）処理完了だ。

さて，多倍長整数どうしの除算になると，処理は急に複雑になる。被除数がm桁，除数がn桁だとすると，商を1桁立てるには，n+1桁の数をn桁の数で割る処理が必要だ。この処理自体，多倍長整数どうしの除算の特殊な場合（商が1桁になることだけわかっている）にすぎず，直接求めるわけにはいかない。実際，人間が筆算で除算を行うときも，除数が2桁以上になると，ある程度の試行錯誤をする。多倍長整数どうしの除算を実現するには，この試行

```
30:		adda.l	d0,a1			*
31:
32:		add.w	d1,d1			*d1 = 被乗数のW単位桁数
33:		subq.w	#1,d1			*dbraを考慮
34:						*X = 0
35:		moveq.l	#0,d3			*最初は繰り上がりなし
36: loop1:	move.w	-(a1),d0		*1桁取り出し
37:		mulu.w	d2,d0			*  乗数を掛けて
38:		addx.w	d3,d0			*  繰り上がり分を足し
39:		move.w	d0,-(a0)		*   1桁分格納
40:		swap.w	d0			*d3 = 繰り上がり
41:		move.w	d0,d3			*
42:		dbra	d1,loop1		*被乗数の桁数分繰り返す
43:
44:		moveq.l	#0,d0			*繰り上がりを処理する
45:		addx.w	d3,d0			*
46:		beq	retn			*
47:		movea.l	RESULT+SAVSIZ(sp),a1
48:		cmpi.w	#BNMAX,(a1)		*
49:		beq	flow			*
50:		move.l	d0,-(a0)		*
51:		addq.w	#1,(a1)			*
52:						*N = 0
53: retn:	movem.l	(sp)+,d0-d3/a0-a1
54:		rts
55:
56: zero:	clr.w	(a0)			*N = 0
57:		bra	retn
58:
59: flow:	moveq.l	#-1,d0			*N = 1
60:		bra	retn
61:
62:		.end
```

リスト7 BNMUL.S

```
 1: *		多倍長整数×多倍長整数
 2:
 3:		.include	bndef.h
 4: *
 5:		.xdef	bnmul
 6:		.xref	bncopy
 7:		.xref	bntemp1
 8:		.xref	bntemp2
 9: *
10:		.offset 4
11: *
12: RESULT:	.ds.l	1		*結果格納領域
13: A:		.ds.l	1		*被乗数
14: B:		.ds.l	1		*乗数
15: *
16:		.text
17:		.even
18: *
19: bnmul:
20:		movem.l	d0-d6/a0-a4,-(sp)
21: SAVSIZ	=	(7+5)*4
22:		movem.l	RESULT+SAVSIZ(sp),a0-a2
23:
24:		move.w	(a1),d1		*被乗数の桁数≧乗数の桁数>0
25:		beq	zero		*  を保証する
26:		move.w	(a2),d2		*
27:		beq	zero		*
28:		cmp.w	d2,d1		*
29:		bcc	noexg		*
30:		exg.l	a1,a2		*
31:		exg.l	d1,d2		*
32:
33: noexg:	cmpa.l	a1,a0		*被乗数と結果格納領域が
34:		bne	nocpy1		*  一致する場合は
35:					*  上書きを避けるため
36:		pea.l	(a1)		*  被乗数をワークにコピーする
37:		lea.l	bntemp1,a1	*
38:		pea.l	(a1)		*
39:		jsr	bncopy		*
40:		addq.l	#8,sp		*
41:
42: nocpy1:	cmpa.l	a2,a0		*乗数と結果格納領域が
43:		bne	nocpy2		*  一致する場合は
44:					*  上書きを避けるため
45:		pea.l	(a2)		*  乗数をワークにコピーする
46:		lea.l	bntemp2,a2	*
47:		pea.l	(a2)		*
48:		jsr	bncopy		*
49:		addq.l	#8,sp		*
50:
51: nocpy2:	move.w	d1,d6		*d6 = 積の桁数（最大値）
52:		add.w	d2,d6		*
53:		bcs	flow		*
54:		cmpi.w	#BNMAX+1,d6	*
55:		bhi	flow		*
56:
57:		moveq.l	#0,d0		*結果格納先を0クリア
58:		move.w	d0,(a0)+	*
59:		move.w	#BNMAX-1,d3	*
60: loop0:	move.l	d0,(a0)+	*
61:		dbra	d3,loop0	*
62:
63:		move.l	#BNVAR,d0	*最下位桁位置を求める
64:		adda.l	d0,a1		*
65:		adda.l	d0,a2		*
```

```
          add.w   d1,d1             *被乗数のW単位桁数
          add.w   d2,d2             *乗数のW単位桁数
          subq.w  #1,d1             *dbraを考慮
          subq.w  #1,d2             *
loop1:    movea.l a1,a4             *a4 = 被乗数最下位桁位置
          moveq.l #0,d4             *被乗数×1桁
          move.w  -(a2),d3          *
          movea.l a0,a3             *
          move.w  d1,d5             *
loop2:    move.w  -(a4),d0          *
          mulu.w  d3,d0             *
          add.l   d4,d0             *
          add.w   d0,-(a3)          *
          swap.w  d0                *
          moveq.l #0,d4             *
          addx.w  d0,d4             *
          dbra    d5,loop2          *
          move.w  d4,-(a3)          *
next1:    subq.l  #2,a0             *桁の重みを考慮
          dbra    d2,loop1          *乗数の桁数分繰り返す
          movea.l RESULT+SAVSIZ(sp),a0
          tst.w   (a0)              *桁数格納フィールドに
          bne     flow              *  積が侵入していたら
                                    *  オーバーフロー
          subq.w  #1,d6             *最上位桁が0なら
          tst.l   (a3)              *  積の桁数は
          beq     done              *  見積もりより1少ない
          addq.w  #1,d6             *
done:     move.w  d6,(a0)           *積の桁数を格納
                                    *N = 0
retn:     movem.l (sp)+,d0-d6/a0-a4
          rts
zero:     clr.w   (a0)              *N = 0
          bra     retn
flow:     moveq.l #-1,d0            *N = 1
          bra     retn
          .end
```

6) 「仮の商」を概算するときの除数の「有効数字」を，2進レベルで最大の桁に取るための処置と思えばよい。たとえば，除数が0001_FFFF$_H$の2^{16}進2桁だったとすると，ただ上位1桁を取り出したのでは，2進での有効数字は1桁しかない。これを15ビット左シフトしてFFFF_8000$_H$にすれば，有効数字が16ビット確保でき，それだけ除算の精度が上がるわけだ。

7) 2進の0.1は10進の0.5，2進の0.01は10進の0.25，以下，2進の小数点以下の各桁の重みは半分になっていく。一般に，n進数の整数部は最下位桁から順にn^0，n^1，n^2，……という重みを持っており，これを下に拡張すれば，小数点以下の各桁はn^{-1}，n^{-2}，……という重みを持つ。

リスト8 BNTEMP1.S

```
          .include        bndef.h
*
          .xdef   bntemp1
*
          .bss
          .even
*
bntemp1:
          .ds.b   BNVAR

          .end
```

リスト9 BNTEMP2.S

```
          .include        bndef.h
*
          .xdef   bntemp2
*
          .bss
          .even
*
bntemp2:
          .ds.b   BNVAR

          .end
```

錯誤をプログラムに取り入れるしかない。要するに，仮に少し大きめの数を商に立ててみて，その数と除数を掛け，被除数(のうち注目している上位桁部分)から引けるかどうかを確かめ，引けなければもう少し小さな数を立ててやり直すわけだ。最初に立てる「仮の商」としては，人間が筆算で除算を行うときのように，被除数の上位数桁を除数の上位数桁で割った値を使えばよい。数桁とはいっても，68000のマシン語ではdivu命令を使うしか手がないから，被除数の上位32ビット÷除数の上位16ビット（2^{16}進でいえば，被乗数の上位2桁÷除数の上位1桁)を「仮の商」として採用することになる。

と，ここまで押さえてもらったところで，プログラム例をリスト11に示そう。リスト11のサブルーチンbndiv中，118～169行がメインループだ。118～121行で被除数の上位2桁を取り出して，除数の上位1桁で割り，「仮の商」を立てる。それから125～137行で作業用領域に除数×「仮の商」を求める。以下，139～145行で被除数との大小関係を比べ，引けるようなら160～165行で実際に引いて，167行で商を立てる。引けなければ149～157行で「仮の商」を1減らし，比較からやり直す。「仮の商」を減らすときには，改めて除数との積を計算せずに，作業用領域に残っている値から除数を引く処理に置き換えて効率を稼いでいる（155～157行)。

リスト11の残りの部分では，56～87行と195～203行に触れておかなければなるまい。56～87行では，除数の最上位ビットが1になるよう，被除数と除数に適当な2^nの値を掛けている。詳しくは触れないが，こうするとdivuで概算する「仮の商」の精度が稼げるのだ[6]。195～203行はこの後始末で，前処理段階で2^n倍してあった分，剰余を修正している。念のためだが，被除数と除数をともに定数倍しても，商は変わらない。

なお，56～57行，195～203行の処理はあるものを利用しただけなのであまり効率はよくない。より真面目にやるなら，多倍長整数のビットシフトをする専用ルーチンを用意すべきだろう。

ここまでで，多倍長整数の四則演算ができるようになったわけだが，せっかく計算できても出力方法がないと困るので（バイナリでダンプすることはできるけど)，リスト12に多倍長整数→10進文字列変換ルーチンbnconvを用意した。通常の，より短い整数同様，10で割った余りを次々に求め，それを逆順にするという方針で作成されている。ただし，1桁出力するごとにbndiv_wで多倍長整数÷16ビット整数の除算を行うのは効率が悪そうだったので，10で割る代わりに10000で割り，その余りを改めて10進4桁に変換するようにしてある。これでbndivwの呼び出し回数は1/4になる。

ついでなので，リスト13に多倍長整数演算ルーチンの簡単なサンプルとして，0！～400！を順に求めるプログラムを示しておこう。

実数演算への応用

さて，多倍長整数演算ルーチンはちょっと工夫すると多倍精度実数演算に応用できる。具体的には，多倍長整数を固定小数点数とみなせばよい。固定小数点数というのは整数部にnビット，小数部にmビットと記憶域を固定した形式の実数データだ。実数値に2^mを掛けて整数化してあると思えばわかりやすいかもしれない。たとえば，2進で，

11.101

のように表される[7]実数値（10進で3.625）は整数部4ビット，小数部4ビットの固定小数点では，

00111010

のように表され（整数だと思えば10進の58），3.625に2^4を掛けて整数化した格好になる。

固定小数点数は場合によっては有効数字の桁数が非常に少なくなってしまうという問題がある[8]ものの，扱いが楽なので限定した場面ではそれなりに有用だ。なんといっても加減算が通常の整数とまったく同じ方法でできるのが大きい。乗除算も整数用の演算ルーチンを流用のうえ，計算後に桁位置を揃えるためにビットシフトすればどうにかなる。

では，多倍精度実数演算のサンプルとして，定番の円周率$\pi=3.14159\cdots\cdots$を求めるプログラムをリスト14に示す。リスト14はπを小数点以下1000桁まで求める。求める値の整数部は10進1桁なので，整数部1桁，残りはすべて小数部という偏った固定小数点形式を採用している。

2進小数から10進文字列への変換にはリスト15に用意した専用ルーチンfixconvを使う。fixconvは1桁の整数部を普通に変換したのち，小数部をどんどん1/10で割って（＝10を掛けて）1桁ずつ10進に変換している。整数の2進→10進変換時には，桁の重みに応じた10^nで次々に割っていくわけだが，小数部にも同じ理屈が通用するのだ。ただし，リスト15のfixconvでは先のbnconv同様に多倍長整数演算の回数を減らす目的で，10000倍して4桁ずつ変換するようにしてある。また，10000倍を繰り返すと下位桁にどんどん0が溜まるため，乗算時にはこの下位桁の0を省くよう細工した。

リスト14ではガウス（Gauss）の公式として知られる，

$$\pi=48\arctan(1/18)+32\arctan(1/57)-20\arctan(1/239)$$

を使ってπを求めている。またarctan[9]の計算には，

$$\arctan(x)=x-x^3/3+x^5/5-x^7/7+\cdots\cdots$$

の級数展開を利用する。この式の各項を順に計算していってアンダーフローした（いま決めた固定小数点形式では表現できなくなって0になってしまった）時点で必要な精度のarctanが得られる。

πの公式としては，このほかにも，マチン（Machin）の公式，

$$\pi=16\arctan(1/5)-4\arctan(1/239)$$

や，Stφrmer（シュテルマー）の公式，

$$\pi=24\arctan(1/8)+8\arctan(1/57)+4\arctan(1/239)$$

などが有名だ。リスト14はわずかな修正でこれらの式用にもなるから試してみるのもいいだろう。

出力桁数はリスト14の8行とそれに応じてBNDEF.H中のBNMAXの値を書き直すことで変更できる。BNMAXをいくつにすればよいかは，必要な小数点以下の10進桁数を$\log_{10} 2^{32}=9.6329\cdots\cdots$で割れば得られる。この際，整数部に1桁余分に必要なことと，若干のゆとり（1桁あれば十分だろう）を設けないと除算時の切り捨てのため，最下位桁に誤差が生じることに注意しよう。

なお，リスト14では固定小数点数を整数で割る際の除数が65535までに制限されている（サブルーチンbndiv_wを使っている）関係で，ある程度以上の精度にはならない。bndivを使うよう変更すればこの制限を取り払うことはできるが，実行速度は数段低下する。また，実行時間は桁数の2乗に比例するので，桁数を増やすときにはそれなりの覚悟がいる。1000桁なら数秒だが，10000桁で6分弱，50000桁では2時間20分ほど，80000桁（このプログラムで求められるほぼ限界の桁数）で6時間かかる。

＊　　　＊　　　＊

さらっと表面を撫でるだけのつもりだったのに，演算ルーチンをワンセット揃えたら結構な分量になってしまった。とはいえ，πのあたりを除けば，どのプログラムも小学校レベルの算数なわけで，何をやっているのか全然わからないということはないと思う。めげずに読み切ってもらいたい。

8）　そのため，一般には有効数字の桁数を常に最大に保つために，必要に応じて整数部と小数部の境界を移動する浮動小数点表現が利用される。浮動小数点表現では数値データの最上位ビットが常に1になるようにして有効数字の桁数を保ち，小数点位置の情報を別に持つ。

9）　arctan（逆正接）はtan（正接）の逆関数で，

$y=\tan(x)$

のとき，

$x=\arctan(y)$

の関係がある。

リスト10　BNDIV_W.S

```
*               多倍長整数÷16ビット整数

                .include        bndef.h
*
                .xdef   bndiv_w
*
                .offset 4
*
RESULT: .ds.l   1               *商
A:      .ds.l   1               *被除数
B:      .ds.w   1               *除数
*
        .text
        .even
*
bndiv_w:
        movem.l d1-d3/a0-a2,-(sp)
SAVSIZ  =       (3+3)*4
        move.w  B+SAVSIZ(sp),d1 *d1 = 除数
        bne     do
                                *除数が0なら
        divu.w  d1,d1           *  故意に0除算して停める

do:     movem.l RESULT+SAVSIZ(sp),a0-a1

        moveq.l #0,d0           *被除数が0なら
        move.w  (a1)+,d2        *
        bne     nzero           *
        move.w  d2,(a0)         *  商も0
        bra     retn            *  剰余も0

nzero:  move.w  #BNMAX,d0       *最上位桁位置を求める
        sub.w   d2,d0           *
        lsl.l   #2,d0           *
        lea.l   2(a0,d0.l),a0   *
        adda.l  d0,a1           *
        movea.l a0,a2           *

        move.w  d2,d3           *d3 = 被除数のL単位桁数
        add.w   d2,d2           *d2 = 被除数のW単位桁数
        subq.w  #1,d2           *dbraを考慮

        moveq.l #0,d0           *最初は余りなし
loop1:  move.w  (a1)+,d0        *上位から1桁取り出し
        divu.w  d1,d0           *  割る
        move.w  d0,(a0)+        *商を立てる
        dbra    d2,loop1        *桁数分繰り返す

        clr.w   d0
        swap.w  d0              *d0.l = 剰余

        subq.w  #1,d3           *上位桁の0を省く
loop2:  tst.l   (a2)+           *
        dbne    d3,loop2        *
        addq.w  #1,d3           *

        movea.l RESULT+SAVSIZ(sp),a0
        move.w  d3,(a0)         *商の桁数を格納

retn:   movem.l (sp)+,d1-d3/a0-a2
        rts

        .end
```

リスト11　BNDIV.S

```
*          多倍長整数÷多倍長整数

           .include          bndef.h
*
           .xdef    bndiv
           .xref    bndiv_w
           .xref    bnadd
           .xref    bncopy
           .xref    bntemp1
           .xref    bntemp2
*
           .offset 4
*
QUOT:      .ds.l    1              *商格納領域
A:         .ds.l    1              *被除数
B:         .ds.l    1              *除数
REM:       .ds.l    1              *剰余格納領域兼ワーク
*
           .text
           .even
*
bndiv:
           movem.l d0-d7/a0-a6,-(sp)
SAVSIZ     =        (8+7)*4
           movem.l QUOT+SAVSIZ(sp),a0-a3

           move.w   (a2),d2               *d2 = 除数桁数
           bne      do                    *除数が0なら
           divu.w   d2,d2                 *  故意に0で除算して停める

do:        pea.l    (a1)                  *被除数を
           pea.l    (a3)                  *  剰余格納先にコピー
           jsr      bncopy                *
           addq.l   #8,sp                 *
           movea.l  a3,a1                 *

           pea.l    (a2)                  *除数を
           lea.l    bntemp1,a4            *  ワークにコピー
           pea.l    (a4)                  *
           jsr      bncopy                *
           addq.l   #8,sp                 *
           movea.l  a4,a2                 *

           move.w   (a1)+,d0              *被除数<除数なら
           cmp.w    d0,d2                 *
           bhi      zero                  *  商は0

           adda.l   #BNVAR,a2             *除数の最上位桁位置を求める
           moveq.l  #0,d4                 *
           move.w   d2,d4                 *
           lsl.l    #2,d4                 *
           suba.l   d4,a2                 *

           add.w    d2,d2                 *d2 = 除数W単位桁数

           cmpi.w   #BNMAX,d0             *除数の最上位ビットが
           beq      noclr                 *  1になるよう
           clr.w    (a1)                  *  除数, 被除数を2^n倍する
noclr:     moveq.l  #1,d1                 *
           move.w   (a2),d0               *
           bne      next0                 *
           addq.l   #2,a2                 *
           subq.w   #1,d2                 *
           subq.l   #2,d4                 *
           move.w   (a2),d0               *
           bra      next0                 *

loop0:     tst.b    (a1)                  *
           bmi      break0                *

           add.w    d1,d1                 *

           pea.l    (a3)                  *
           pea.l    (a3)                  *
           pea.l    (a3)                  *
           jsr      bnadd                 *

           pea.l    (a4)                  *
           pea.l    (a4)                  *
           pea.l    (a4)                  *
           jsr      bnadd                 *

           lea.l    12+12(sp),sp          *

           add.w    d0,d0                 *
next0:     bpl      loop0                 *
break0:    move.w   d1,-(sp)              *何倍したか覚えておく

           move.w   (a2),d3               *d3 = 除数最上位w

           move.w   (a3)+,d1              *被除数の最上位桁位置を求める
           move.l   #BNMAX,d0             *
           sub.w    d1,d0                 *
           lsl.l    #2,d0                 *
           adda.l   d0,a1                 *

           add.w    d1,d1                 *商の最上位桁位置を求める
           sub.w    d2,d1                 *
           moveq.l  #0,d0                 *
           move.w   d1,d0                 *
           add.l    d0,d0                 *
           adda.l   #BNVAR-2,a0           *
           suba.l   d0,a0                 *

           move.w   d1,-(sp)              *商のW単位桁数-1を待避
           move.l   a0,-(sp)              *商の最上位桁位置を待避

           clr.w    -2(a0)                *商格納先と被除数の
           clr.w    -2(a1)                *  上位半桁分をクリアしておく

           lea.l    0(a1,d4.l),a3         *(a1)～(a3) ... 被除数の上位桁
           lea.l    0(a2,d4.l),a4         *(a2)～(a4) ... 除数
           lea.l    bntemp2,a5            *(a5)～(a6) ... 除数×仮の商を
           lea.l    2(a5,d4.l),a6         *  求めるための作業用領域

           moveq.l  #0,d0                 *最初は余りなし
           bra      lpent
loop1:     move.w   (a1)+,d0              *被除数の上位2wを
           swap.w   d0                    *  除数の上位1wで割り
lpent:     move.w   (a1),d0               *  商にいくつが立つか
           divu.w   d3,d0                 *  概算する
           tst.w    d0                    *
           beq      next1                 *1すら立たなかった

           movem.l  a4/a6,-(sp)           *求めた概算値を
           moveq.l  #0,d6                 *  除数に掛けて
           move.w   d2,d7                 *  被除数から引く値を
           subq.w   #1,d7                 *  求める
loop2:     move.w   -(a4),d5              *
           mulu.w   d0,d5                 *
           add.l    d6,d5                 *
           move.w   d5,-(a6)              *
           swap.w   d5                    *
           move.w   d5,d6                 *
           dbra     d7,loop2              *
           move.w   d6,-(a6)              *
           movem.l  (sp)+,a4/a6           *

retry:     movem.l  a1/a5,-(sp)           *引けるか?
           subq.l   #2,a1                 *
                                          *
           move.w   d2,d7                 *
loop4:     cmpm.w   (a5)+,(a1)+           *
           dbne     d7,loop4              *
           movem.l  (sp)+,a1/a5           *

           bcc      subit                 *引ける

           subq.w   #1,d0                 *仮の商が大きすぎたので
           beq      next1                 *  1減らしてやり直す
                                          *
           movem.l  a4/a6,-(sp)           *
           sub.w    d7,d7                 *
           move.w   d2,d7                 *
loop5:     subx.w   -(a4),-(a6)           *
           dbra     d7,loop5              *
           movem.l  (sp)+,a4/a6           *
           bra      retry                 *

subit:     movem.l  a3/a6,-(sp)           *引く
           sub.w    d7,d7                 *
           move.w   d2,d7                 *
loop6:     subx.w   -(a6),-(a3)           *
           dbra     d7,loop6              *
           movem.l  (sp)+,a3/a5           *

next1:     move.w   d0,(a0)+              *商を1桁立てる
           addq.l   #2,a3                 *桁の重みを考慮
           dbra     d1,loop1              *必要な桁数分繰り返す

           movea.l  (sp)+,a0              *a0 = 商の最上位桁
           move.w   (sp)+,d0              *d0 = 商の仮桁数
loop7:     tst.l    (a0)+                 *商の上位桁の0を省く
           dbne     d0,loop7              *
           addq.w   #1,d0                 *

           movea.l  QUOT+SAVSIZ+2(sp),a0
           move.w   d0,(a0)               *商の桁数を格納

           movea.l  REM+SAVSIZ+2(sp),a0
           movea.l  a0,a1                 *剰余の上位桁の0を省く
           adda.l   #BNVAR,a1             *
           suba.l   d4,a1                 *
           lsr.w    #1,d2                 *
           bcc      nodd                  *
           clr.w    -(a1)                 *
           addq.w   #1,d2                 *
nodd:      subq.w   #1,d2                 *
loop8:     tst.l    (a1)+                 *
           dbne     d2,loop8              *
           addq.w   #1,d2                 *

           move.w   d2,(a0)               *剰余の桁数を格納

           move.w   (sp)+,d0              *初期化段階で
           cmpi.w   #1,d0                 *  定数倍していたら
           beq      retn                  *  その分剰余を補正する

           move.w   d0,-(sp)
           pea.l    (a0)
           pea.l    (a0)
           jsr      bndiv_w
           lea.l    10(sp),sp

retn:      movem.l  (sp)+,d0-d7/a0-a6
           rts

zero:      clr.w    (a0)                  *
           bra      retn                  *

           .end
```

▶ 前からおかしかった内蔵の20Mバイトハードディスクがとうとう壊れてしまいました。せっかくインストールした日本語LaTexシステムがぁ！　でも,これで素直にSCSIへ移行できそうです。とほほ……。
猿山　拓路(21)

リスト12　BNCONV.S

```
*         多倍長整数→文字列変換

          .include        bndef.h
*
          .xdef     bnconv
          .xref     bndiv_w
          .xref     bncopy
          .xref     bntemp1
*
          .offset 4
*
STRBUF:   .ds.l     1             *結果格納領域
A:        .ds.l     1             *変換対象
*
          .text
          .even
*
bnconv:
          movem.l d0-d1/a0-a2,-(sp)
SAVSIZ    =         (2+3)*4
          movem.l STRBUF+SAVSIZ(sp),a0-a1

          lea.l     bntemp1,a2          *変換対象を
          pea.l     (a1)                *  ワークにコピー
          pea.l     (a2)                *
          jsr       bncopy              *

          movea.l a0,a1
          move.w    #10000,-(sp)        *下位から10進4桁ずつ
loop1:    pea.l     (a2)                *  取り出して
          pea.l     (a2)                *
          jsr       bndiv_w             *
          addq.l    #8,sp               *

          moveq.l #3-1,d1                         *  1桁ずつ
loop2:    divu.w    #10,d0                        *  10進文字列に変換
          swap.w    d0                            *
          addi.b    #'0',d0                       *
          move.b    d0,(a0)+                      *
          clr.w     d0                            *
          swap.w    d0                            *
          dbra      d1,loop2                      *
          addi.b    #'0',d0                       *
          move.b    d0,(a0)+                      *

          tst.w     (a2)                          *0になるまで
          bne       loop1                         *  繰り返す

          moveq.l #'0',d0                         *上位桁の'0'を省く
          moveq.l #3-1,d1                         *
loop3:    cmp.b     -(a0),d0                      *
          dbne      d1,loop3                      *
          beq       skip                          *
          addq.l    #1,a0                         *

skip:     sf.b      (a0)                          *文字列終端

loop4:    move.b    (a1),d0                       *文字の並びが
          move.b    -(a0),(a1)+                   *  逆になっているので
          move.b    d0,(a0)                       *  反転する
          cmpa.l    a0,a1                         *
          bcs       loop4                         *

          lea.l     8+2(sp),sp
          movem.l (sp)+,d0-d1/a0-a2
          rts

          .end
```

リスト13　BNFACT.S

```
*         階乗を求める

          .include        doscall.mac
          .include        const.h
          .include        bndef.h
*
          .xref     bnmul_w
          .xref     bnconv
*
          .text
          .even
*
ent:
          lea.l     inisp,sp

          lea.l     varA,a0             *A = 1
          move.w    #1,(a0)             *
          move.l    #1,varA+BNVAR-4     *

          moveq.l #1,d6                 *B = 1

          move.w    #100-1,d7
loop:     bsr       print

          move.w    d6,-(sp)            *A = A * B
          pea.l     (a0)                *
          pea.l     (a0)                *
          jsr       bnmul_w             *
          lea.l     10(sp),sp           *
          bmi       flow                *

          addq.w    #1,d6               *B++
          dbcs      d7,loop
          bcs       flow

          bsr       print
          DOS       _EXIT
*
print:
          pea.l     (a0)                          *10進出力
          pea.l     strbuf                        *
          jsr       bnconv                        *
          DOS       _PRINT                        *
          pea.l     crlf(pc)                      *
          DOS       _PRINT                        *
          lea.l     12(sp),sp                     *
          rts
*
flow:     pea.l     errmes(pc)
          DOS       _PRINT

          DOS       _EXIT
*
errmes:   .dc.b     'オーバーフローが発生しました'
crlf:     .dc.b     CR,LF,0
*
          .bss
          .even
*
varA:     .ds.b     BNVAR                         *n!
strbuf:   .ds.b     BNMAX*10+10                   *10進文字列格納用

          .stack
          .even
*
          .ds.l     256
inisp:

          .end      ent
```

リスト14　BNPI.S

```
*         円周率πを求める (Gaussの公式)

          .include        doscall.mac
          .include        iocscall.mac
          .include        bndef.h
          .include        const.h
*
MAXDEC    equ       1000      *小数点以下10進桁数
*
          .xref     bnadd
          .xref     bnsub
          .xref     bndiv_w
          .xref     fixconv
*
          .text
          .even
*
ent:
          lea.l     inisp,sp

          lea.l     pi,a0
          clr.w     (a0)
          lea.l     temp1,a1
          lea.l     temp2,a2

          lea.l     bnadd,a3
          lea.l     bnsub,a4
          lea.l     bndiv_w,a5

          moveq.l #'a',d1
          move.w    #BNMAX,(a1)                   *+48arctan(1/18)
          move.w    #48,4(a1)                     *
          move.w    #18,d2                        *
          move.w    #18*18,d3                     *
          bsr       atan                          *

```

▶高校生活は短いなあー。ついこの間，部活を引退しました。あと半年でほぼ終わりですね。このひまな時間を使ってプログラミングしようかなぁ。　　美辺　央希(18)東京都

```
        moveq.l #'b',d1
        move.w  #BNMAX,(a1)     *+32arctan(1/57)
        move.w  #32,4(a1)       *
        move.w  #57,d2          *
        move.w  #57*57,d3       *
        bsr     atan            *

        moveq.l #'c',d1
        exg.l   a3,a4           *-20arctan(1/239)
        move.w  #BNMAX,(a1)     *
        move.w  #20,4(a1)       *
        move.w  #239,d2         *
        move.w  #239*239,d3     *
        bsr     atan            *

        bsr     print

        DOS     _EXIT
*
atan:
        movea.l sp,a6

        move.w  d2,-(sp)
        pea.l   (a1)
        pea.l   (a1)
        jsr     (a5)            *div_w
        move.w  d3,8(sp)

        pea.l   (a0)
        pea.l   (a0)
        jsr     (a3)            *add / sub
        addq.l  #8,sp

        moveq.l #3,d4
loop:   jsr     (a5)            *div_w
        tst.w   (a1)
        beq     retn
        move.w  d4,-(sp)
        pea.l   (a1)
        pea.l   (a2)
        jsr     (a5)            *div_w
        pea.l   (a0)
        pea.l   (a0)
        jsr     (a4)            *sub / add
        lea.l   10+8(sp);sp
        addq.w  #2,d4
        bcs     flow

        jsr     (a5)            *div_w
        tst.w   (a1)
        beq     retn
        move.w  d4,-(sp)
        pea.l   (a1)
        pea.l   (a2)
        jsr     (a5)            *div_w
        pea.l   (a0)
        pea.l   (a0)
        jsr     (a3)            *add / sub
        lea.l   10+8(sp),sp
        addq.w  #2,d4
        bcs     flow

        IOCS    _B_PUTC
        bra     loop

flow:   moveq.l #'X',d1
        IOCS    _B_PUTC
retn:   moveq.l #CR,d1
        IOCS    _B_PUTC
        moveq.l #LF,d1
        IOCS    _B_PUTC

        movea.l a6,sp
        rts
*
print:
        move.l  #MAXDEC,-(sp)
        pea.l   (a0)
        pea.l   strbuf
        jsr     fixconv
        DOS     _PRINT

        pea.l   crlf(pc)
        DOS     _PRINT

        lea.l   12+4(sp),sp
        rts
*
crlf:   .dc.b   CR,LF,0
*
        .bss
        .even
*
pi:     .ds.b   BNVAR           *π
temp1:  .ds.b   BNVAR           *x^(2n+1)
temp2:  .ds.b   BNVAR           *x^(2n+1)/(2n+1)
strbuf: .ds.b   MAXDEC+10

        .stack
        .even

        .ds.l   1024
inisp:

        .end    ent
```

リスト15 FIXCONV.S

```
*       多倍精度固定小数点実数→文字列変換

        .include        bndef.h
*
        .xdef   fixconv
        .xref   bncopy
        .xref   divu32
        .xref   bntemp1
*
        .offset 4
*
STRBUF: .ds.l   1       *結果格納領域
A:      .ds.l   1       *変換対象
NDIGIT: .ds.l   1       *小数点以下最大桁数
                        *(4の倍数であること)
*
        .text
        .even
*
fixconv:
        movem.l d0-d5/a0-a4,-(sp)
SAVSIZ  =       (6+5)*4
        movem.l STRBUF+SAVSIZ(sp),a0-a2
        move.l  a2,d3
        lsr.l   #2,d3           *d3 = 桁数/4
        subq.l  #1,d3           *dbraを考慮
        bcs     done

        lea.l   bntemp1,a2      *変換対象を
        pea.l   (a1)            *  ワークに転送
        pea.l   (a2)            *
        jsr     bncopy          *
        addq.l  #8,sp           *

        move.l  #BNMAX,d0       *最上位桁位置を求める
        move.w  (a2)+,d4        *
        sub.w   d4,d0           *
        lsl.l   #2,d0           *
        lea.l   0(a2,d0.l),a1   *
        add.w   d4,d4           *d4 = W単位の桁数

        movea.l a0,a3           *整数部を10進変換
        moveq.l #0,d0           *
        cmpa.l  a1,a2           *
        bcs     loop0           *
        move.l  (a1)+,d0        *
        subq.w  #2,d4           *
loop0:  moveq.l #10,d1          *
        jsr     divu32          *
        addi.b  #'0',d1         *
        move.b  d1,(a0)+        *
        tst.l   d0              *
        bne     loop0           *
                                *
        move.l  a0,d1           *
loop1:  move.b  (a3),d0         *
        move.b  -(a0),(a3)+     *
        move.b  d0,(a0)         *
        cmpa.l  a0,a3           *
        bcs     loop1           *
        movea.l d1,a0           *

        move.b  #'.',(a0)+      *小数点

        subq.w  #1,d4           *小数部がなければ終了
        beq     done            *

        move.l  #BNVAR-2,d0
        lea.l   0(a2,d0.l),a3   *a3 = 最下位桁位置直後

        move.w  #10000,d1
        addq.l  #2,a2

        swap.w  d3
loop2:  swap.w  d3
loop3:
loop4:  tst.w   -(a3)           *下位桁の0を省く
        dbne    d4,loop4        *
        beq     break           *
        addq.l  #2,a3           *

loop5:  tst.w   (a1)+           *上位桁の0を省く
        dbne    d4,loop5        *
        beq     break           *

        movea.l a3,a1           *値を10000倍
```

```
        moveq.l #0,d2               *
        move.w  d4,d5               *
loop6:  move.w  -(a1),d0            *
        mulu.w  d1,d0               *
        add.l   d2,d0               *
        move.w  d0,(a1)             *
        moveq.l #0,d2               *
        swap.w  d0                  *
        move.w  d0,d2               *
        dbra    d5,loop6            *
        addq.w  #1,d4               *
        move.w  d2,-(a1)            *

        moveq.l #0,d0               *10000倍後の整数部を
        cmpa.l  a1,a2               * 10進変換
        bcs     skip                *
        move.w  (a1)+,d0            *
        subq.w  #1,d4               *
skip:   lea.l   exptbl(pc),a4       *
        moveq.l #3-1,d5             *
loop7:  divu.w  (a4)+,d0            *
        addi.b  #'0',d0             *
        move.b  d0,(a0)+            *
        clr.w   d0                  *
        swap.w  d0                  *
        dbra    d5,loop7            *
        addi.b  #'0',d0             *
        move.b  d0,(a0)+            *

        dbra    d3,loop3
        swap.w  d3
        dbra    d3,loop2

break:  cmpi.b  #'.',-1(a0)
        beq     done

        moveq.l #3-1,d1             *下位桁の'0'を省く
        moveq.l #'0',d0             *
loop8:  cmp.b   -(a0),d0            *
        dbne    d1,loop8            *
        beq     done                *
        addq.l  #1,a0               *
done:   sf.b    (a0)                *

        movem.l (sp)+,d0-d5/a0-a4
        rts
*
exptbl: .dc.w   1000,100,10

        .end
```

表1　10^nの命数法

n	日	米	英
0	一 ［いち］	one	one
1	十 ［じゅう］	ten	ten
2	百 ［ひゃく］	hundred	hunderd
3	千 ［せん］	thousand	thousand
4	万 ［まん］		
6		million	million
8	億 ［おく］		
9		billion	milliard
12	兆 ［ちょう］	trillion	billion
15		quadrillion	
16	京 ［けい］		
18		quintillion	trillion
20	垓 ［がい］		
21		sextillion	
24	杼 ［じょ］	septillion	quadrillion
27		octillion	
28	穣 ［じょう］		
30		nonillion	quintillion
32	溝 ［こう］		
33		decillion	
36	澗 ［かん］	undecillion	sextillion
39		duodecillion	
40	正 ［せい］		
42		tredecillion	septillion
44	載 ［さい］		
45		quattuordecillion	
48	極 ［ごく］	quindecillion	octillion
51		sexdecillion	
52	恒河沙 ［ごうがしゃ(こうがしゃ)］		
54		septendecillion	nonillion
56	阿僧祇 ［あそうぎ］		
57		octodecillion	
60	那由他（那由多）［なゆた］	novemdecillion	decillion
63		vigintillio	n
64	不可思議 ［ふかしぎ］		
66			undecillion
68	無量(大)数 ［むりょう(たい)すう］		
72			duodecillion
78			tredecillion
84			quattuordecillion
90			quindecillion
96			sexdecillion
102			septendecillion
108			octodecillion
114			novemdecillion
120			vigintillion
303		centillion	
600			centillion

表2　10^{-n}の命数法

n	
−1	分 ［ぶん］
−2	厘 ［りん］
−3	毛 ［もう］（毫［ごう］）
−4	糸(絲) ［し］
−5	忽 ［こつ］
−6	微 ［び］
−7	繊 ［せん］
−8	沙 ［しゃ］
−9	塵 ［じん］
−10	埃 ［あい］
−11	渺 ［びょう］
−12	漠 ［ばく］
−13	模糊 ［もこ］
−14	逡巡 ［しゅんじゅん］
−15	須臾 ［しゅゆ］
−16	瞬息 ［しゅんそく］
−17	弾指 ［だんし］
−18	刹那 ［せつな］
−19	六徳 ［りっとく］
−20	虚 ［きょ］
−21	空 ［くう］
−22	清 ［せい］
−23	浄 ［じょう］

大数・小数の読み方

一，十，百，千，万，億，兆，京。現在使われている数はせいぜいこのくらいまでだ。が，もっと大きな数にもちゃんと表1のような名前がついている。この命数法は江戸時代に書かれた日本最初の数学書『塵劫記（じんこうき）』に記されている。

ついでだったので，表には米・英での大数の呼び方も一緒に並べておいた。見比べてみると，日本語では4桁進法，米・英では3桁進法が採用されているという目立った相違に加えて，思想・文化の違いも見て取れてなかなか面白い。日本の命数法には恒河沙，阿僧祇，那由他のような梵語に由来する語が含まれるなど仏教の影響が強く表れているのに対して，米・英の命数法に宗教色はなく，ひたすら合理的にできている。billion以降はmillionの2乗，3乗，4乗，……，という意味での命名であり，2，3，4，……を表す接頭語“bi-”，“tri-”，“quadri-”などmillionとの合成語にすぎない。ただし，米語では10の3乗単位になってしまっているので元の意味は失われている。

さて，『塵劫記』には小さな数の命数法も記されていて，こちらは表2のようになっている。西洋では15世紀まで小数の概念がなかったそうで，そのためか小数を表す数詞はない。

なお，歩合や比率を表すときにも表2と同様の言葉を用いるが，その場合は“割”が基本で，“分”はその10分の1（だから1/100），“厘”はそのまた10分の1（だから1/1000）というように，表とは1桁ずれる。

以上，本筋とはあまり関係のない雑学であった（読者が自発的に多倍長整数の日本語表記出力ルーチンを作ったりすると，少しは関係ある話になるわけだけど）。

愛読者プレゼント

シャープ ☎03(3260)1161

1 Easypaint SX-68K

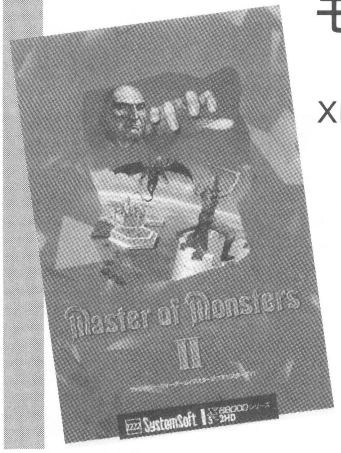
X68000用 3.5/5"2HD版

12,800円(税別) 3名

まずはSX-WINDOW対応のグラフィックツール「Easypaint SX-68K」をプレゼント。SX-WINDOW ver.1.10以上が必要です。

システムソフト ☎092(752)5278

2 マスターオブモンスターズII



X68000用 5"2HD版3枚組

9,800円(税別)

3名

"大戦略"シリーズで有名なシステムソフトが放った，ファンタジックシミュレーションゲーム。魔法と剣で戦ってください。

ポニーキャニオン ☎03(3221)3161

3 ダライアス CD

1,500円(税込) 2名

9月号の善バビで紹介した，タイトーのアーケードゲーム「ダライアス」の復刻版CD。オリジナル曲に加え，アレンジバージョンも収録。

集英社 ☎03(3230)6191

4 UFOキャッチャー公式ガイドブック

450円(税込) 2名

もう流行は過ぎ去ったかもしれないけど，UFOキャッチャーはいまだに人気者。でも，"公式"っていわれるとなんか笑ってしまう。

5 ヨンセンマン消しゴム

日産自動車4000万台突破記念の広告でお馴染み，ヨンセンマンの消ゴムをプレゼント。これはある人が車を買ってもらったものです。

ヨンセンマン消しゴム 3名

プレゼントの応募方法

とじ込みのアンケートハガキの該当項目をすべてご記入のうえ，希望するプレゼント番号をハガキ右下のスペースにひとつ記入してお申し込みください。締め切りは1992年10月18日の到着分までとします。当選者の発表は1992年12月号で行います。

8月号プレゼント当選者

1 シムアース (東京都)萩原今卓肥 (大阪府)今井彰彦 2 スピンディジーII (京都府)松尾文人 (和歌山県)藤井時光 (岡山県)三宅良和 3 X68000マシン語プログラミング グラフィック編 (北海道)深井康行 (山形県)工藤憲和 (東京都)後藤芳史 (神奈川県)森田剛 (岡山県)山下善之 4 飲み物他詰め合わせ (兵庫県)佐尾和博 (敬称略)

以上の方々が当選しました。おめでとうございます。商品は順次発送いたしますが，入荷状況などにより遅れる場合もあります。また，雑誌公正競争規約の定めにより，当選された方はこの号の他の懸賞には当選できない場合がありますのでご了承ください。

郵便はがき

料金受取人払

高輪局承認

1396

差出有効期間
平成6年7月
15日まで

1 0 8 - 0 0

5 0 7

(受取人)

東京都港区高輪
2-19-13 NS高輪ビル

ソフトバンク株式会社

Oh!X 編集部行

□□□-□□　電話

住所

氏名　年齢

職業・勤務先
学校・学部・学年

●編集部へのメッセージ

今月号の特集について

いちばん良かった記事	興味のなかった記事
これから載せてほしい記事内容	本誌以外にお読みのパソコン雑誌

推薦する市販ソフト

ソフト名：

推薦理由：

あなたにとって音楽とはなんですか？

あなたの愛機は(所有機種に○印をつけてください)　ない

X1(マニアタイプ,C,D,F,G,twin) X1 turbo(model 10,20,30,40,II,III,Z,ZII,ZIII)

MZ-(80K/C, 1200, 700, 1500, 80B, 2000, 2200, 2500, 2861)

X68000(初代,ACE,PRO,PROII,EXPERT,EXPERTII,SUPER,XVI,Compact, HD)

その他　MIDI楽器(　)

FD(　基)　TAPE　QD　HD(　MB)　MO　プリンタ(　)

年齢　歳　パソコン歴　年　男・女　プレゼントNo.

切り取り線

通常払込料金
加入者負担

払込通知票

口座番号	東京	1	29307	金額	
加入者名	ソフトバンク株式会社			料金	払込み / 特殊 円
払込人住所氏名	※ （郵便番号 ）			備考	
				受付局日付印	

各票の※印欄は、払込人において記載してください。

記載事項を訂正した場合は、その箇所に訂正印を押してください。

この払込通知票は、機械で使用しますので、下部の欄を汚さないよう特に御注意ください。また、本票を折り曲げたりしないでください。（郵政省）

切り取らないで郵便局にお出しください。

通常払込料金
加入者負担

払込票

口座番号	東京	1	29307
加入者名	ソフトバンク株式会社		
金額			
払込人住所氏名	※		
備考		受付局日付印	

振替用紙

←点線から、きれいに切り取ってご使用ねがいます。

切り取り線

【定期購読のご案内】

●定期購読のお申し込みは、この郵便振替用紙のみとさせて頂き、銀行振込・現金書留によるご入金は、ご遠慮下さい。

●受付締切は、

1日発売	：	発売日前月10日振込
8日発売	：	〃 15日振込
15・18日発売	：	〃 25日振込

です。

〈例〉 4月1日発売（Oh！PC 4月15日号）の場合、お振込の締切は3月10日です。

締切に間に合わなかった場合は、自動的に次号からの発送となります。

なお、すでに発売されているもの、また、お振込が締切に間に合わなかった月号のものは、定期購読ではお求めになれません。書店でご購入ください。

●定期購読誌のお届けは書店発売日より遅くなりますのでご了承下さい。

「発売日一覧」

◇毎月1・15日発売
Oh！PC

◇毎月18日発売
Oh！X
Oh！FM TOWNS
C MAGAZINE
Oh！Dyna

◇毎月8日発売
月刊情報処理試験
LANTIMES
THE WINDOWS
DOS／Vmagazine
UNIX USER

切り取り線

切り取らないで郵便局にお出しください。

この欄は、加入者あての通信にお使いください。

送り先

お名前 フリガナ　　　お電話

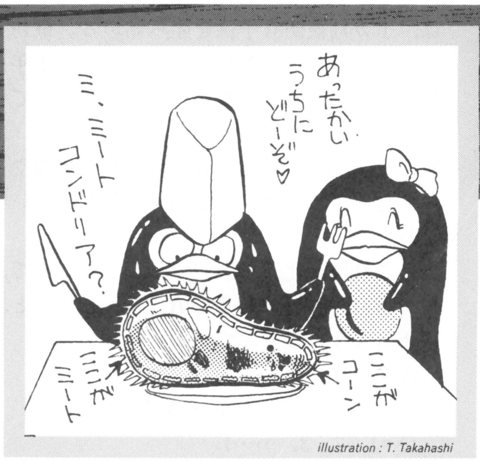
ご住所 フリガナ
〒

定期購読申込書

□Oh！PC	年間	(23回)	12,880円	(新規／継続 NO.)
□Oh！PC	6ヶ月	(12回)	6,720円	(新規／継続 NO.)
□Oh！X	年間	(12回)	7,200円	(新規／継続 NO.)
□Oh！FM TOWNS	年間	(12回)	7,440円	(新規／継続 NO.)
□C MAGAZINE	年間	(12回)	11,760円	(新規／継続 NO.)
□Oh！Dyna	年間	(12回)	9,120円	(新規／継続 NO.)
□月刊情報処理試験	年間	(12回)	8,160円	(新規／継続 NO.)
□月刊情報処理試験	6ヶ月	(6回)	4,080円	(新規／継続 NO.)
□LANTIMES	年間	(12回)	17,760円	(新規／継続 NO.)
□THE WINDOWS	年間	(12回)	11,760円	(新規／継続 NO.)
□DOS／Vmagazine	年間	(12回)	9,360円	(新規／継続 NO.)
□UNIX USER	年間	(12回)	11,760円	(新規／継続 NO.)

〔備考欄〕

この払込通知票は、機械で使用しますので、下部の欄を汚さないよう特に御注意ください。また、本票を折り曲げたりしないでください。 （郵政省）



マシン語カクテル
in Z80's Bar
第36回

仕様の違いはなんのその，強引に作ったCASLtoZ80ソースコンバータ。今回はサンプルプログラムを用意して，サポートしきれなかったアドレス操作について光君が具体的に説明していきますが，おや？　こんなところに小さなバグが……。

お城と流れ星 ―その3―

Kaneko Shunichi　金子 俊一

カランコロ～ン♪
源光（以下光）：こんにちは。
ようこ（以下Yo）：いらっしゃいませ。
マスター(以下M)：うっわ～，焼けてますね。
長老(以下老)：おおかた日焼けサロンにでも通ったんじゃろ。
光：違いますよ。
M：きっと，ディスプレイの前に座りすぎて……。
老：ディスプレイ焼けか？
光：そんな不健康な。
Yo：やっぱり左手の先に注目するべきよね。
老：少し白いのう。きっとサンオイルを左手だけで塗ったせいじゃろう。右手も使えば，こんな情けないことにはならなかったじゃろうにな。
光：そんなわけありませんよ。
M：まさか，ゴルフ焼け？
光：そうですよ。だいたい日焼けの話題なんて，10月号としては不適切なんじゃないんですか。9月号でも会ってるわけだし。
Yo：スポーツの秋ってことにすればいいじゃないの。
老：プログラムの秋って言葉もあるぞ。
光：僕は聞いたことないですよ。
M：食欲の秋ってことで何か食べます？
光：マスターのお勧めでいいですよ。
Yo：えっと，今日はアメリカ風のドリアがお勧めよ。
光：じゃあそれと，炎のバルセロナ。
老：おお～っ。勝利をめざす炭酸飲料，炎のバルセロナがあるのか，この店は。
M：ええ，ポッカの品物も扱ってますからね。
Yo：どんな味がするの？
M：ちょっと炭酸を弱くしたオロナミンCですね。
老：それが350ml缶に入っておるのじゃ。デカビタCなぞ目じゃないわい。
Yo：なるほど。それで勝利をめざす炭酸飲料なのね。

CASL's Bar

光：やっぱり，今月も僕がプログラミングするんでしょうね。
老：何をわかりきったことをいっておるのじゃ。
Yo：今月は何を作るの？
光：えっと，レジスタを表示するプログラムとか，実際にCASLのプログラムをいくつか作ってみようと思っているんですけど。
Yo：さすがに考えているのね。
M：Z80's BarというよりCASL's Barだな，こりゃ。
光：大丈夫ですよ，レジスタを表示するプログラムはZ80で書きますから。
Yo：ほかにはどんなプログラムを作ってくれるの？
光：えっとですね，Oh!MZ1986年12月号に出ていた，CASL&COMETのサンプルプログラムを実行させてみようと。
老：そういえば7月号で純ちゃんがやっていた第1種情報処理技術者試験のプログラムは動くんかいのう。
光：えっと，変更なしで動くかどうかはリストを見てみないとなんともいえません。
M：ほい，7月号。
光：え～と，このプログラムは変更の必要がありそうだな。
Yo：それじゃあ，まとめてやってちょうだいね。
光：ほ～い。

先月号にバグ発見

光：カチャカチャ……げげっ！
Yo：仮面ライダーアマゾンの腕輪がどうしたの？
老：ほっほっほ。それは「ギギ」の腕輪じゃ。ちょっと高度なボケじゃのう。
Yo：いやいや。
M：で，光君はどうしたの？
光：いや実は……。
老：おおかた先月号のプログラムにバグでも見つけたんじゃろうて。
光：お恥ずかしい。とりあえずパッチを当てるプログラムも作っときますから。
Yo：掻かずに，ぱっち。
老：今日はボケが絶好調だのう。
M：それで，どんなバグなんですか？
光：CALL文の解釈がおかしくなっていて，変換後のソースリストにラベル名が入らないんですよ。
M：原因はわかったんですか。
光：ええ，結構いきあたりばったりで作ってたんで，サブルーチンのバージョンアップの際にちょっと仕様を変更してしまったんですよ。
老：それが命取りになったわけじゃな。
光：申しわけないですだ。
老：そんなボーンヘッドをやらかすとは，光，地に落ちたな。
光：はっはっは。まだほかにもあるかもしれませんけどね。
Yo：あんたが開き直ってどーすんのよ。

嗚呼！　ミトコンドリア

光：カチャカチャ……でーきたっと。
M：ほい，こちらもできましたよ。ミートコーンドリアお待たせ。
光：ミトコンドリア!?
Yo：ミートコーンドリア。お肉ととうもろこしを使ったアメリカ風のドリアよ。
老：うむ，確かに毛唐は肉ととうもろこしで暮らしておるわい。
光：すいません，ちょっと食欲が。

M：やっぱ，ネーミングがよくなかったかなあ。

Yo：ところでプログラムは？

光：まず，先月号のバグを取るパッチプログラムがこれ（リスト1）です。

老：どーすればいいんじゃ。

光：先月号のオブジェクトプログラムをロードしておいてから，リスト1をアセンブルして，8000_H番地を実行するだけです。

Yo：実行後はセーブするのね。

光：基本です。

Yo：そういえば先月号で予告していた，そのままでも動くプログラムと変更の必要があるプログラムの解説は？

光：これからちゃんとやります。

M：とりあえずどんなプログラムを作ったのか教えてくださいよ。

光：えっとCASLとしてはOh!MZ1986年12月号のサンプルとして載っていたフィボナッチ数列のプログラムと，1992年の第1種情報処理技術者試験のプログラムですね。

老：予定どおりになったわけじゃな。

光：ええ。それからCASL&COMETのGR0～GR4とFR（フラグレジスタ）を表示するプログラムも作りました（リスト2）。

Yo：これでレジスタの状態が見やすくなったわね。

M：使い方は？

光：コンバートしたプログラムを実行後，リスト2の実行アドレス（$E000_H$番地）をコールするだけです。

老：画面に表示されないレジスタ内容の結果を表示するわけじゃな。

光：そうです。

フィボナッチのうさぎ

光：まず，変更なしで動くプログラムの代表として，フィボナッチ数列のプログラムを見てみましょう（リスト3）。

Yo：ねえねえ，そのセコビッチ数列って何者よ。

光：（無視）聞かれると思ってました。

M：ちゃんと調べたんですね。

光：ええ，フィボナッチが1202年に書いた算術の本の中にあった問題なんです。「1つがいのうさぎは，毎月1つがいずつ子供を産む。新しく産まれたうさぎは，1カ月後から子供を産み始める。最初に1つがいのうさぎがいたとすると，1年後には何つがいになるか」ってやつなんですよ。

Yo：それって，平成教育委員会が25時間やってたときに，ヒツジで同じ問題が出ていたわ。

光：これは漸化式で表記することができて，Nカ月目のつがいの数をA［N］とすると，以下のようになるんですよ。

A[1]=1，A[2]=1，

A[N+2]=A[N+1]+A[N]

(N=1，2，3，……)

老：さっぱりわからん。

M：同じく。

Yo：以下同文。

光：結局，答えは1，1，2，3，5，8，13，21，……って具合になるんですけどね。

Yo：ふ～ん。それはいいけど，なんでこのプログラムは変更なしで動くのよ。

光：それはあとで説明しますね。

老：ところで，このプログラムを実行すると，どうも答えが間違っておるようなのじゃが。

光：ええ，そのようですね。これは1986年12月号のバグですけど。

老：間違いないんか。おぬしのプログラムがとんでもないコンバートをしておるんじゃないのか？

光：そんなことはないです。元のプログラムのアルゴリズムが違っていますよ。

Yo：なんだかな。

M：ちゃんとしたフィボナッチ数列を求めるプログラムも作ってあるんでしょ。

光：リスト4です。

M：使い方も教えていただけると嬉しいんですけど。

リスト1

```
0000            ;   KAKAZU_NI_PATCH
0000            ;
0000            ;       by Hikaru Minamoto
0000
8000                    ORG     $8000
8000
8000            RETXASS EQU     $D187
8000            LTNL$   EQU     $D610
8000            COPY$   EQU     $D5BF
8000            TRNS$   EQU     $D615
8000
8000 21 BE DF           LD      HL,L_COPY$
8003 22 07 D2           LD      ($D207),HL ; Patch !
8006                    ;
8006 21 C7 DF           LD      HL,START_P
8009 22 17 D1           LD      ($D117),HL ; Patch !
800C                    ;
800C C9                 RET
800D
800D
DFBE                    ORG     $DFBE
DFBE            L_COPY$
DFBE 1A                 LD      A,(DE)
DFBF 77                 LD      (HL),A
DFC0 13                 INC     DE
DFC1 23                 INC     HL
DFC2 FE 0D              CP      $0D
DFC4 20 F8              JR      NZ,L_COPY$
DFC6 C9                 RET
DFC7            START_P
DFC7 CD 10 D6           CALL    LTNL$
DFCA C3 87 D1           JP      RETXASS
OBJECT CODE END DFCC
```

リスト2

```
0000            ;   CASL REG. PRINT
0000            ;
0000            ;       by Hikaru Minamoto
0000
E000                    ORG     $E000
E000
E000            #PRTHL  EQU     $1FBE   ; AF
E000            #PRTHX  EQU     $1FC1   ; AF
E000            #MPRINT EQU     $1FE2   ; AF,DE
E000            #LTNL   EQU     $1FEE   ; nothing
E000            #PRINT  EQU     $1FF4   ; F
E000            GR0     EQU     $DF00
E000            FR      EQU     $DF0A
E000
E000            START
E000 06 00              LD      B,0
E002            LOOP
E002 CD E2 1F           CALL    #MPRINT
E005 47 52              DM      "GR"
E007 00                 DS      1
E008                    ;
E008 3E 30              LD      A,$30
E00A 80                 ADD     A,B
E00B CD F4 1F           CALL    #PRINT
E00E                    ;
E00E CD E2 1F           CALL    #MPRINT
E011 20 3D 20           DM      " = $"
E014 24
E015 00                 DS      1
E016                    ;
E016 11 00 DF           LD      DE,GR0
E019 26 00              LD      H,0
E01B 68                 LD      L,B
E01C 29                 ADD     HL,HL
E01D 19                 ADD     HL,DE
E01E 5E                 LD      E,(HL)
E01F 23                 INC     HL
E020 56                 LD      D,(HL)
E021 EB                 EX      DE,HL
E022 CD BE 1F           CALL    #PRTHL
E025 CD EE 1F           CALL    #LTNL
E028                    ;
E028 3E 05              LD      A,5
E02A 04                 INC     B
E02B B8                 CP      B
E02C 20 D4              JR      NZ,LOOP
E02E                    ;
E02E CD E2 1F           CALL    #MPRINT
E031 46 52 20           DM      "FR  = "
E034 20 3D 20
E037 00                 DS      1
E038                    ;
E038 21 0A DF           LD      HL,FR
E03B 7E                 LD      A,(HL)
E03C CD C1 1F           CALL    #PRTHX
E03F CD EE 1F           CALL    #LTNL
E042 C9                 RET
OBJECT CODE END E042
```

光：えっと，リスト3,4ともメインルーチンの最初でGR1にNカ月目のNを入れるわけですよ。

Yo：このプログラムでは８カ月目になっているわけね。

光：そうです。あとはコンバータを使ってZ80のプログラムにコンバートして，アセンブラでアセンブルすれば実行できます。答えはGR2レジスタに入ってきますから，レジスタを表示するプログラムを使って，結果を確認してみてください。

老：どれどれ１年後は何つがいになっておるのかな。

M：なんと144つがいですか。

老：これではすぐにも65535つがいを超えてしまいそうじゃな。

光：ええ，33カ月目までは計算できますが，34カ月目でオーバーフローを起こします。

Yo：０カ月目とか試しちゃいけないのね。

光：さっきの漸化式の条件に(N＝1, 2, 3, ……)っていうのがあったでしょ。

老：リスト３のほうは再起呼び出しを使っておるようじゃが，実行時間が長いのう。暴走してるかと思ったぞ。

光：ええ。答えは違うし，時間はかかるし，あんまりいいとこないみたいですね。

2分木

光：それではお待ちかね，変更しなければ動かないプログラムいってみましょうか。

Yo：第１種情報処理技術者試験のプログラムね（リスト５）。

光：図１を見てください。

老：いかにも２分木じゃな。

光：ええ。この図でいえば，１つひとつのマス目が１ワードになっていて，最初のマ

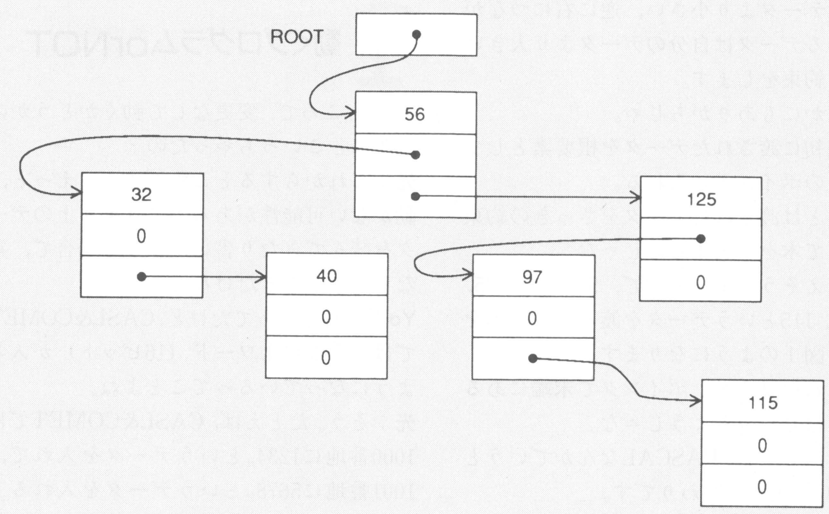
リスト3

```
;   Fibonacci   86'12
;
;        by Hikaru Minamoto

MAIN    START
        LEA     GR1,8
        CALL    FNUMB
        RET

FNUMB   START
        CPA     GR1,CONST3
        JPZ     NEXT
        LEA     GR2,-1,GR1
        RET
NEXT    PUSH    0,GR1
        LEA     GR1,-1,GR1
        CALL    FNUMB
        PUSH    0,GR2
        LEA     GR1,-1,GR1
        CALL    FNUMB
        ST      GR1,WRK
        POP     GR2
        ADD     GR2,WRK
        POP     GR1
        RET
CONST3  DC      3
WRK     DS      1
        END
```

リスト4

```
;   Fibonacci  true
;
;        by Hikaru Minamoto

MAIN    START
        LEA     GR1,8
        PUSH    0,GR1
        CALL    INIT
        CALL    FNUMB
        POP     GR1
        RET

INIT    START
        LEA     GR0,1
        ST      GR0,A_WRK
        LEA     GR0,0
        ST      GR0,B_WRK
        RET

FNUMB   START
        LEA     GR1,-1,GR1
        JNZ     NEXT
        LD      GR2,A_WRK
        RET
NEXT    LD      GR2,A_WRK
        ADD     GR2,B_WRK
        ST      GR2,A_WRK
        SUB     GR2,B_WRK
        ST      GR2,B_WRK
        JMP     FNUMB
A_WRK   DS      1
B_WRK   DS      1
        END
```

図１　2分木の例



リスト5

```
;    BTREE
;
;        by Hikaru Minamoto

MAIN    START
        LEA     GR2,DATA
MAINLP
        PUSH    0,GR2
        LEA     GR1,ROOT
        CALL    BTREE
        POP     GR2
        LEA     GR2,6,GR2
        CPL     GR2,DATEND
        JNZ     MAINLP
        END
ROOT
        DS      1
DATA
        DC      56
        DS      2
        DC      32
        DS      2
        DC      125
        DS      2
        DC      40
        DS      2
        DC      97
        DS      2
        DC      115
        DS      2
DATEND
        DC      DATEND

BTREE   START
        PUSH    0,GR1
        PUSH    0,GR3
        LD      GR3,0,GR1
        LEA     GR3,0,GR3
        JZE     STORE
LOOP    LD      GR1,0,GR3
        CPA     GR1,0,GR2
        JMI     RIGHT
        JZE     ERROR
        LEA     GR1,2,GR3    ; Z80 ARRANGE
        JMP     NEXT
RIGHT   LEA     GR1,4,GR3    ; Z80 ARRANGE
NEXT    LD      GR3,0,GR1
        LEA     GR3,0,GR3
        JNZ     LOOP
STORE   ST      GR2,0,GR1
        LEA     GR0,0
        JMP     FIN
ERROR   LEA     GR0,1
FIN     POP     GR3
        POP     GR1
        RET
        END
```

スがデータ，次のマスは左へのポインタ，その次は右へのポインタになっています。

老：ポインタにはアドレスが入るのかな？

光：そうなります。そこで，左へのポインタの先につながれているすべてのデータは自分のデータより小さい，逆に右につながっているデータは自分のデータより大きいという約束をします。

老：いかにもありがちじゃ。

光：最初に渡されたデータを根要素としてルートのポインタに入れる。

老：あとは渡されたデータをさっきの約束に従って木を生成するのじゃな。

光：ええそうです。そこで，56，32，125，40，97，115というデータを渡して２分木を作ると図１のようになります。

老：これを見ると，ポインタで末端にあるところは0が入るようじゃな。

光：そうですね。PASCALなんかでいうところの「nul」の代わりです。

M：このプログラムの実行方法は？

光：こちらも単にコンバートしてアセンブルするだけです。

M：結果はどのように確認すればいいんですか。

光：データのアドレスを見て，ポインタがちゃんと図１と同じようにつながっているかを確認するんです。

Yo：これって，データを変更すればもっといろんな２分木も作れるのよね。

光：そうですね。ただし，このままではデータの削除ができないので，そこいらへんも対応させたほうが面白くなると思いますよ。

Yo：ふ～ん。

動くプログラムorNOT

Yo：ところで，変更なしで動くかどうかの説明はどこいっちゃったの？

光：これからするところですよ。えっと，動かない可能性があるのはメモリ上のデータを読んできたり書いたりする場合で，重なりがあるときだけなんですよ。

Yo：先月もいってたけど，CASL&COMETでは１番地に１ワード（16ビット）が入るようになっているってことよね。

光：そう。たとえば，CASL&COMETでは1000番地に1234_Hというデータを入れて，1001番地に5678_Hというデータを入れると図２のようなメモリ構成になる。

Yo：なんか違和感を感じるよね。

老：いままでZ80だけを扱ってきたから，よけいそう感じるのじゃろう。

光：Z80では同じことをすると図３のようになる。

Yo：つまり，上書きされてデータが変わってしまうのね。

光：こういった場合に動かなくなってしまうんだ。

Yo：なるほど。

光：そこで，アドレス計算をずらしたりして，重ならないようにプログラムの変更が必要になってくる。

老：さっきのフィボナッチはメモリにはアクセスしておったが，特定のアドレスでデータが重ならなかったため，変更なしで動いたわけじゃな。

光：ええ。

Yo：２分木のほうはテーブルみたいな構造になっていて，アドレス計算をしていたから，そこを変更する必要があったってことね。

光：ええ，こちらも変更点が２カ所あっただけで完全に動きました。

*　　*　　*

Yo：どうもごくろうさま。

光：CASL&COMETを勉強するなら，もっとちゃんとしたエミュレータを作ったほうがいいんでしょうね。

Yo：でもとりあえず動くわけだし。

老：動かないプログラムだって変更すればいいんじゃしのう。

M：まあせっかくだからミートコーンドリアでも食べて。

光：どれどれ……うっリボゾームの味が。

Yo：するわきゃないだろ！

—つづく—

図2　COMETでのメモリアクセス例

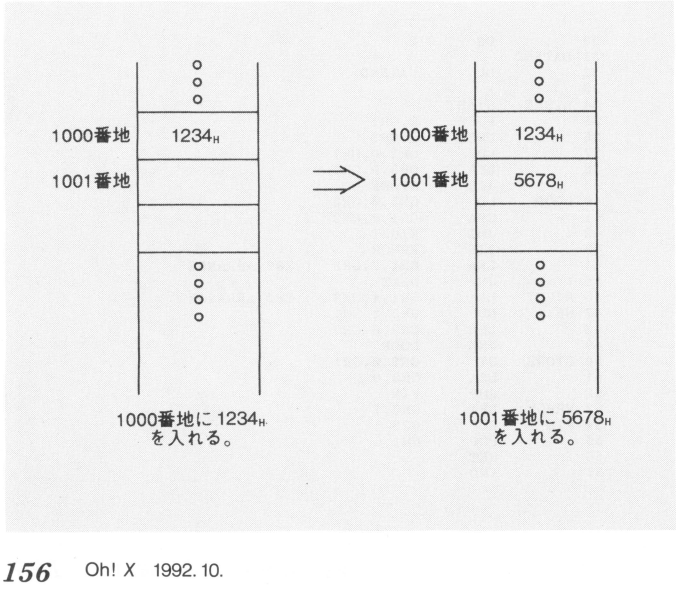
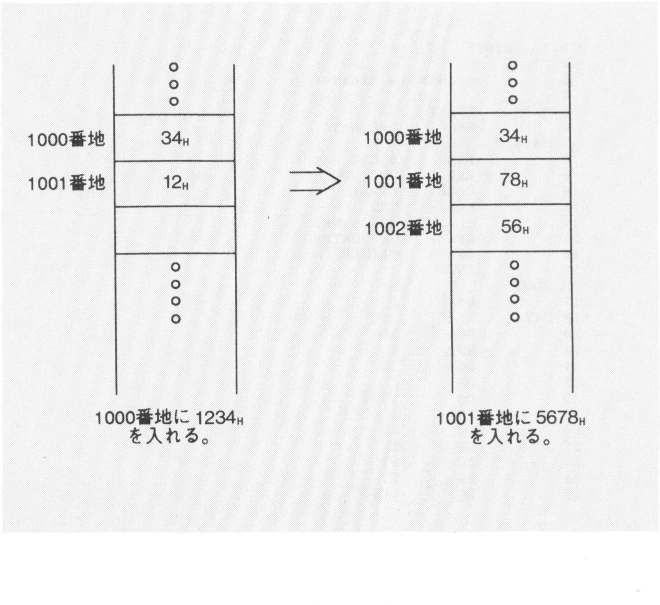


図3　Z80でのメモリアクセス例



Creative Computer Music入門(13)

ボーカルのかわりに……

予定をちょっと早めて今回から応用編に入ります。最初はボーカル付きの曲をインストゥルメンタルにアレンジするときにどうすればいいかを考えてみましょう。歌声はコンピュータミュージックでは扱いにくいテーマです。これまでの知識を総動員して読んでみてください。

Taki Yasushi 瀧 康史

§ 2年目を迎えて

担：いままで理論ばっかりやってたでしょ。1カ月休んでいいから，2年目はすぐに役に立つ応用編やってよ。
瀧：う〜ん，たとえば？
担：それはあなたが考えるんじゃない？
瀧：そんなあ。う〜ん，サビで盛り上げる方法，その1（かけ上がり）とかぁ？　それに1回休んじゃうのもなんだし（車の積み立てもあるし）。とりあえず，今月は1年間の復習ってのは？
U：もともと，いろいろ端折りながら，まとめられるだけまとめてきたんだから，これ以上まとめられます？
瀧：じゃあ，何がいいかな〜。
U：たとえば，ボーカルがある曲をコンピュータミュージックに直すとか。あのテの曲はコンピュータミュージックにすると，しっくりこないでしょう？
瀧：う〜みゅ，具体的。でも，内容は漠然としてる。んじゃ，いろいろ考えてみようかな……。

＊　＊　＊

という具合で，今月のテーマはボーカルのある曲（歌）をどのように処理するか，ということになりました。テーマもテーマで，漠然としているため，いままでよりもさらに「絶対的な理論」が通じません。そのため，かなりとりとめのない話や，漠然とした定理が出てくると思うので，さっと流すか，真剣に考え込むかどちらかにしてください。中途半端に考えると余計にわからなくなりますよ，ってことです。

考えてみれば，この連載も進行方向をまじめに考えねばならない時期に立たされています。できるかぎり読者の立場になって決めたいのも山々ですが，まず基礎を固めなければ，応用知識も実用にはならないというのも真実であって，なかなかむずかしい選択でもあります。

でも，何の役に立つかわからない知識を蓄えるより，音楽は実践。馬鹿のひとつ覚えでもいいから，やってみるが勝ち。結局は，それが私の信条。

馬鹿のひとつ覚えって言葉は嫌いですけど，覚えたことを使って何が悪い。得意げになって何が悪いってね。

まあ，馬鹿が云々はどうでもいいんですが，ここはとりあえず入門編を終えて，1年蓄えた知識を下敷きにして，実践編へ進むことにしましょう。できるかぎり，“これは第○回でやった，××理論の応用だよ”ということを伝えながら，進行していきたいと思っています。

§ 歌の歴史的な背景

えっと，蘊蓄（うんちく）が嫌いな方はとばしてくれて結構。まずはここで，音楽の主流なるものを考えてみましょう。ロック？　ジャズ？　フュージョン？　いえいえ，そんな新しいものではなく，もっと長い目で見つめ直してみてください。

いま，皆さんにシンフォニーをイメージしてといったら，個人的な相違はあるでしょうが，壮大な世界，雄大なリズム，分厚い音，こみ上げる感動……。まあ，いろいろ思いつくでしょうが，昔から音楽の中心的なもので，優遇されてきたと感じている（考えているではないぞ）かと思います。

しかし，皆さんがよく知っているシンフォニー（オーケストラによるもの）は，最初から音楽の主役だったわけではありませんでした。中世のコンサート（音楽会）というのはオペラ（歌）を聴くためのもので，そこにはたしかにオーケストラもいましたが単なる伴奏にすぎず，単独で演奏するのは前座，幕間，歌手が退場したあとの終楽章という程度でした。

当時は“コンサートをやるのか，それで歌手は誰？”といったような感じで，現在のように，どこのオーケストラだとか，指揮者は誰だとかいうことに聴衆の関心はなかったのです（指揮者が現在のように指揮棒を振るようになったのは近代になってからです）。

このシンフォニーを歌なしでも聴けるものへと確立させたのがモーツァルトといえるのですが，超有名な彼の作ったシンフォニーでも，その当時に注目を浴びるのはなかなかたいへんなことだったのです。

じゃあ，それ以前や，あるいはもっと違う地域ではどうだったんでしょう。

原始の時代から，人はものを叩いてリズムを刻むとともに，やはり歌って（うなって？）いたのではないでしょうか。いちばん手近にあって，容易に使える楽器，それが歌声であることは間違いありません。

いつの時代でも，人の声に優る楽器はなく，楽器は人の声に負けまいとして，発展してきたといっても過言ではありません。

ここでの問題はその逆，その歌をどのようになくすかということですから，この作業が容易でないことは，簡単に理解できますよね？

§ なぜ，貧弱になるか

先に述べたとおり，人間の声は実にさまざまな表情を持っています。これとはほかに，歌詞（言葉）は人に理解してもらえるということも挙げられます。

言葉には意味がある，このことは音楽に表情をつけ，歌詞を通して，人々の心に，作詞者，歌手，演奏者，または作曲者の心を伝えやすくします。それらが“一般的”で“人々の心を捕らえやすい”要因なのですが，実はこの利点は逆にメロディに対しての甘さを招きやすいのです。

たとえば，ただ同じ音が8分音符でズラ〜っとつながってるようなメロディでも，それなりの歌詞をつければ十分にさまになります。

例はいっぱい転がってますよね。たまたま思いついた，「風の谷のナウシカ」のテーマソングでも，サビでさえ単音が「カ・ゼ・ノ・タ・ニ」と5連できます。人の声があ

まりにも繊細で表情があり，また，聴衆は曲ではなく歌詞を聴きがちなので，それなりのメロディとして落ち着きますが，これがピアノ，極端な話，FM音源のちゃちなピアノなら，ただ同じ音が5つ鳴るだけの，つまらないものになってしまいます。生楽器ならなんとか工夫もできますが，コンピュータサウンドでは単調さに拍車をかけることにもなりかねません。

おわかりのとおり，歌なら，メロディがいいかげんでも詞と人間の歌声がその穴埋めをやってくれるのです。

歌をコンピュータミュージックにすると，変になってしまう大きな理由は，“もとのメロディの貧弱さをカバーできない”ことにあるのではないでしょうか？

§ どのように克服するか

変になる理由がわかったのですから，これを逆手にとりその欠点を補う，もしくはそうならないようにうまく逃げる解決法を見つけてみましょう。

極端な話，同じ音が（しかもサビで）ただズラッと並ぶ曲は使わなければいいのです。原曲をしっかり聴いてみて，それが人の声ではなく，貧弱で色彩のないシンセサイザの音でもおかしくはならないか？　そういった点に注意して，選曲をすればいいのです。終わり。

……でも，やっぱり好きな曲をやりたいですよね？　ここで終わったらそれこそ，怒りの封書が送られてきそうなので，それなりにほかの面から攻めてみましょう。

当然ですが，そういう曲（ただ同じ音が連なるような）はアレンジするのに，大きなハンデがあることを覚悟しておいてください。しかし，それをなんとかしてうまくコンピュータミュージック化するのもまた一興でしょう。

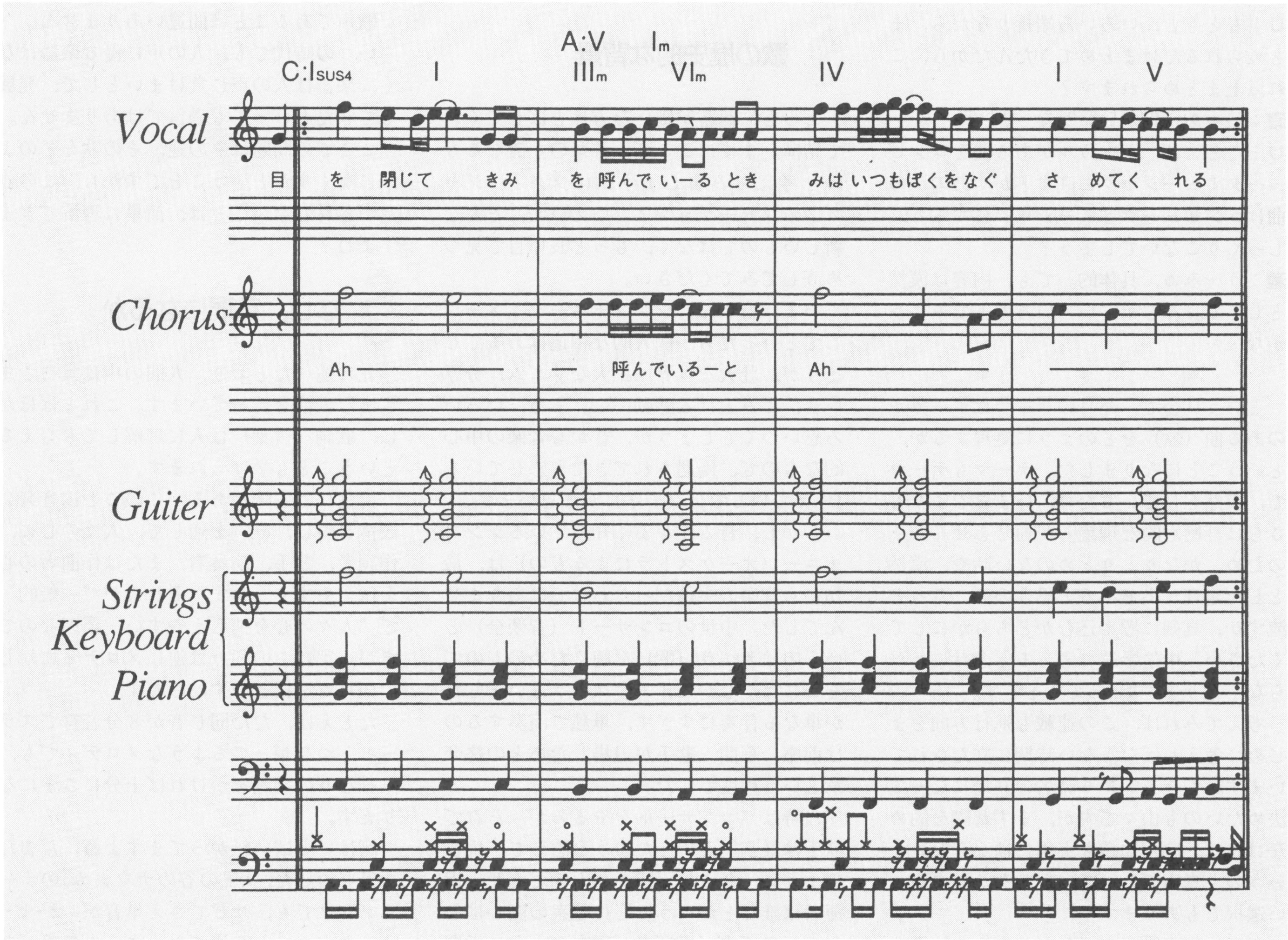
とりあえず，私が使う方法を整理してみましょう。メロディが単調にしろ，そうでないにしろ，歌をインストゥルメンタルにするときには，当然ながらある程度のアレンジが必要となります。この理由は何度もいったとおり，歌声と楽器の音には表現力において大きな隔たりがあることです。

たとえば，楽譜1のようなメロディがあったとしましょう。私が大昔に作ったバンド用の曲からピックアップしたものです。

もともと原曲は女性ボーカルが歌ったものなので，音域の高いG(maj)スケールだったのですが，わかりやすくするために，C(maj)に移調しています。このくらいの音域だと，声の高いテナーの男性ならば歌える音程ですね。

楽譜1をMML化したのがリスト1です。歌のパートもすべてそのまま，比較的感じの似ているCM-64の内蔵音色に割り当てているだけです。ボーカルのパートは，イメージ的に音色カード6のクラリネットがよかったのですが，あんまりカードばかり使っていると，パンチがとんできそうなので，LA音源のクラリネットを使いました。カードを買うだけですむ人（CM-64）ならいいのですが，カード買っても意味がない場合（CM-500など）もありますからね。

CM-64では，ギターのナチュラルトーンの音がベロシティが低いとミュートしてし

楽譜1

まうので，アコースティックギターの音を使用，オクターブとベロシティを上げて，その分，ヴォリュームを下げています。

そのため，一部のコンパチ音源では，バランスが多少狂うかもしれません。しかし，そのほかは音を素直に使っているので，まあ聴けるレベルでしょう（というか，これが満足に聴けなかったら，コンパチとはいえないはず)。

さて，まさかアマチュアマイナーバンドのライブを聴きにきたことがある人はいないでしょうから，誰も聴いたことがない歌を題材として使っていることになります。

この曲のメロディはどちらかというと動くほうですので，歌つきの原曲を聴いてなければあまり変には聴こえないかもしません。あえていうなら，歌詞に合わせるために音が細切れになるところ（あとで説明します)，スラーのかかったところ。まあ，そのあたりじゃないかと思います。

すでに気がついているとは思いますが，よく聴く歌には思い入れがあります。しかし，インストゥルメンタルにするときにアレンジされ，曲自身が変わってしまうのはよくあることです。まずは聴く側も，原曲と同じじゃなくちゃだめだとか，そういった狭い聴き方を改める必要があるでしょう。少しぐらいの違いよりも，それが曲として完成されているかどうかが問題なのです。

うちのバンドの連中にも，カラオケの伴奏を聴きながら，ドラムパートが違ったり，ギターパートが違ったりすると，「これ違う」とか，とやかくいう人がいます。たしかに，なかには聴くに耐えないものもあるのは認めますが，そういった曲もそれなりに聴いてみれば，面白いところに原曲にはないオブリガードがあったり，間奏を短くするために極端なアレンジをしてあったりして，なかなか楽しめますよ。

§ 曲の解説など

コード進行は，これまでに耳にタコができるほど解説したからわかりますよね。代理コードなども入っていますが，特別おかしな進行もなく，ごく基本的ですから，疑問は出ないはずです。出ないはずだと思います。出ないといいなぁ（自分の教え方に対して弱気になってしもうた)。

ま，疑問点を感じたら，1992年1月号や9月号を中心に復習してください。

曲の構成は，これまでこの講座を読み続けてきた人には簡単だと思います。少なくとも，「バナナ」や「ゼクセクス」よりは単純で簡単です。

ドラムパートは私が作ったんじゃなくて，エゴの強いドラマーの気合が入ったアレンジのせいで，多少自己陶酔的な面があるかもしれません。ウラが山のようにありますが，強拍はシンバルなどで押さえていますから，耳が肥えてない人でも，さして変には聴こえないでしょう（多少暴れてる感じはあるけど)。

そのほかのパートはごくありきたりです。バラードにありがちなギターナチュラルトーンも，データ中ではディレイ&リバーブがかかっていませんが，本来ならこのパートにだけかけるとよろしいです。和音に遅めのディレイをかけて，アルペジオっぽい感じを出しています。

注目すべきは，ここのコードの構成音なのですが，ギターパートだけでは2音（とそのオクターバ）しか，コードの構成音が含まれていません。これはギターの，特にポップス系によくある（というかほとんどそうです）パワーコードというやつで，3度の音を省略してしまうという，まさしく力技コードなのです。こうすると，メジャー，マイナー，それにサスフォーなどの区別がなくなってしまうので，どちらでも同じコードを弾いてかまわないという策略です（要はラクしてるんです。きっと)。

たとえば，Cならコードの構成音のEを省略し，CG<CGというふうに，作り変えてしまいます。ポップスギターはCEGと密着して鳴らすよりも，CGCと鳴らしたほうが一般的です。また，この場合のように，ナチュラルトーンなら普通にCEGとコードを鳴らしてもかまわないのですが，ディストーションギターなどでは，音が比較的こもりやすいので，パワーコードのほうがよりリアルになるでしょう。

また，お決まりの経過的うんぬんってのがありまして，パワーコードどうしの連結である構成音が経過的に続くなら，3度の音を使ってみるのもいいでしょう。もうひとつのコードの構成音は保続がかかるといいですね。

キーボードパートはありがちで，片方がピアノのコード，もう片方がストリングスです。見れば見るほど簡単なつくり。また，よくあるパターンのひとつに右コード左ベースノートってのもありますが，この曲のベースラインはベースノートから動かない(!)のでこういうつくりになっています。

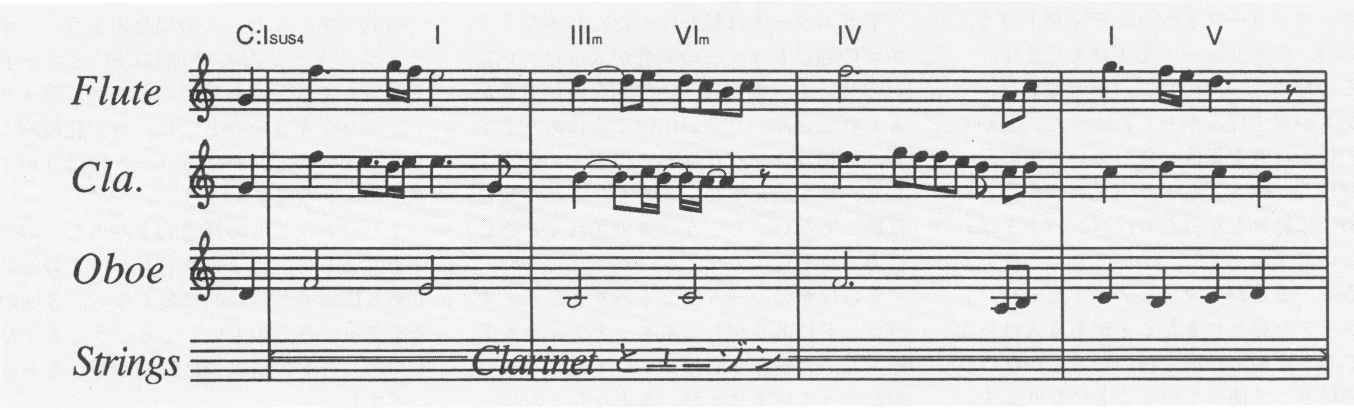
§ アレンジしなくちゃ

さて，楽譜1&リスト1が原曲そのままだとしたら，楽譜2&リスト2は，この曲をインストゥルメンタルとしてアレンジし直したものです。

楽譜2はたったの4段しかありませんが，楽譜1のギターのナチュラルトーンとキーボードのピアノパート，そしてベースとドラムを加えて考えてください。リスト2は

リスト1

```
.comment 「忘れたくない友達のために」からサビの1部(原曲)
.comment  作詞/作曲 紅呪威羅黄爵神    アレンジ Go&幻想皇帝
.comment  Roland CM64とそのコンパチ用 MML化 瀧 康史

(i)      /       初期化コマンド

(bl)     /       MIDI優先モード

(m1,2000) (aMIDI02,1)   / Vocal    女性のボーカルが好ましい。
(m2,2000) (aMIDI11,2)   / Chorus   一応、男性ファルセット。
(m3,2000) (aMIDI12,3)   / Guiter   ナチュラルトーン&ディレイ
(m4,2000) (aMIDI13,4)   / Keyboard ストリングス
(m5,2000) (aMIDI14,5)   / Keyboard ピアノ
(m6,2000) (aMIDI15,6)   / Bass     エレキベース
(m7,2000) (aMIDI10,7)   / Drum     手1シンバル系(左右手じゃないよ)
(m8,2000) (aMIDI10,8)   / Drum     手2スネア、タム系
(m9,2000) (aMIDI10,9)   / Drum     足

(o70)

(t1) @84 @v110 @u090 L16 o4 g4 <f4e8de&e4.>gg   b4bb<c>b&baa4a8
(t2) @31 @v085 @u060 L16 o5 r4 f2e2             d4dd e d&dcc4r8
(t3) @11 @v075 @u120 L02 o4 r4 'cg<cg',8'cg<cg' >'b<eb<e''a<ea<e'
(t4) @36 @v085 @u050 L02 o5 r4 fe               >b<c
(t5) @05 @v105 @u060 L04 o4 r4 'g<cf''g<cf''g<ce''g<ce'
(t5)                                  'gb<e''gb<e''a<ce''a<ce'
(t6) @24 @v090 @u080 L04 o2 r4 cccc             >bbaa
(t7) @r  @v110 @u110 L08 o2 r4 <c#4>@u80f#4f#f#a#4
(t7)                                            a#4f#4f#f#a#4
(t8) @r  @v110 @u110 L08 o2 r4 r4d4ddd4         d4d4ddd4
(t9) @r  @v110 @u110 L16 o2 r4 c8.crcrcrcrcrrcc rcrcrcrcrcrcrrcc

(t1) <f8f8ffgf&fed8c8d8 c4c8dc&c>bb4.
(t2) f2>a4a8b8          <c4>b4<c4d4
(t3) <'cf<cf''cf<cf'    'cg<cg'>'b<gb<g'<
(t4) f.>a4<             c4>b4<c4d4
(t5) 'a<cf''a<cf''a<cf''a<cf'
(t5)                    'g<ce''g<ce''gb<d''gb<d'
(t6) ffff               L8g4rgagabL4
(t7) a#f#f#f#r16f#16ra#r@u110<c#rc#r>@u80f#f#rr
(t8) rrdrr16d16rdrL16   rdrrdrrdrdr@u127~20<c32c32c32r32>bg_20L8@u110
(t9) crrcrrcccrrcrrcc   crcccrccrcrcrrrr
(p)                /* 演奏命令
```

楽譜2

リスト2

```
.comment 「忘れたくない友達のために」からサビの1部
.comment  作詞/作曲 紅呪威羅黄爵神   アレンジ Go&幻想皇帝
.comment   インストゥルメンタルへのアレンジ   瀧 康史
.comment  Roland CM64とそのコンパチ用 MML化 瀧 康史

(i)     /       初期化コマンド

(b1)    /       MIDI優先モード

(m1,2000) (aMIDI02,1)   / Flute
(m2,2000) (aMIDI03,2)   / Clarinet
(m3,2000) (aMIDI04,3)   / Oble
(m4,2000) (aMIDI11,4)   / Guiter(Natural Tone with Delay.)
(m5,2000) (aMIDI13,5)   / String
(m6,2000) (aMIDI15,6)   / Bass     エレキベース
(m7,2000) (aMIDI10,7)   / Drum     手1シンバル系(左右手じゃないよ)
(m8,2000) (aMIDI10,8)   / Drum     手2スネア、タム系
(m9,2000) (aMIDI10,9)   / Drum     足
(m10,2000) (aMIDI14,10) / Keyboard ピアノ

(o70)

(t1) @73 @v110 @u080 L16 o5 g4 f4.gfe2             d4.e8d8c8>b8<c8
(t2) @83 @v120 @u110 L16 o3 g4 @84<f4e8de&e4.>g8 b4.<c>b&ba&a4r8
(t3) @85 @v085 @u060 L02 o5 d4 fe                  >b<c
(t4) @11 @v075 @u120 L02 o4 r4 'cg<cg',8'cg<cg' >'b<eb<e''a<ea<e'
(t5) @36 @v095 @u064 L16 o5 g4 f4.gfe2             d4.e8d8c8>b8<c8
(t10) @05 @v105 @u060 L04 o4 r4 'g<cf''g<cf''g<ce''g<ce'
(t10)                                 'gb<e''gb<e''a<ce''a<ce'
(t6) @24 @v090 @u080 L04 o2 r4 cccc               >bbaa
(t7) @r  @v110 @u110 L08 o2 r4 <c#4>@u80f#4f#f#a#4
(t7)                                             a#4f#4f#f#a#4
(t8) @r  @v110 @u110 L08 o2 r4 r4d4ddd4          d4d4ddd4
(t9) @r  @v110 @u110 L16 o2 r4 c8.crcrcrcrcrrcc rcrcrcrcrcrrcc

(t1) f2.>a8<c8            @u+10g4.@u-10fed4.r8
(t2) <f4.gf&fed8c8d8     c4d4c4>b4
(t3) f.>a8b8             <c4>b4<c4d4
(t4) <'cf<cf''cf<cf'     'cg<cg'>'b<gb<g'<
(t5) f2.>a8<c8            @u+10g4.@u-10fed4.r8
(t10) 'a<cf''a<cf''a<cf''a<cf'
(t10)                     'g<ce''g<ce''gb<d''gb<d'
(t6) ffff                L8g4rgagabL4
(t7) a#f#f#f#r16f#16ra#r@u110<c#rc#r>@u80f#f#rr
(t8) rrdrr16d16rdrL16    rdrrrdrrdrdr@u127~20<c32c32c32r32>bg_20L8@u110
(t9) crrcrrcccrrcrrcc    crcccrccrcrcrrrr
(p)              /* 演奏命令
```

これらのパートがすべて入ったかたちで掲載されていますが，打ち込むときはリスト1をいじるようにすれば，それほど苦労せず入力できるでしょう。

楽譜2と楽譜1の大きな違いはメロディパートです。原曲は3年ほど前に作ったものなので，ほかにもいじりたいところがたくさんあるのですが，それだと単なるインストゥルメンタル化じゃなくて，違うジャンルへのアレンジになりそうなので，メロディのみのアレンジにしました。流れるピアノとか，ブラスの合いの手とか入れたかったんですけどね。決して手抜きとかじゃないわよ！

楽譜1を見ればわかると思うのですが，先ほど少し話したとおり，歌のメロディは台詞に合わせるため，こまめに音を切る感があります。そこで2連の16分音符などを，8分音符ひとつにまとめていきます。それをつきつめていくと，一般の楽器でも演奏できるものに変化していきます。

問題になる音色の選び方ですが，ここでは歌の代わりをやる楽器にはクラリネット，コーラスとしてフルート，そして，それらの補佐としてオーボエを入れています。

一般にインストゥルメンタルへのアレンジでは，このように木管系の楽器や，ブラス系（特にサックスなど）を使うことが多いようです。この曲はどちらかというとバラードですが，バラードでもジャズっぽい曲はサックスがよく似合います。ただし，音の響きやうねりに十分注意して使わないとダサくなります。逆になんでも合ってしまうのが，木管＋弦のユニゾンです。

コーラス部のアレンジはもはや原形をとどめておらず，ほとんどオブリガードになっています。狙いは流れるようなコーラスパートをオブリガードにしたかったこと，単音ではメロディパートが薄っぺらになるので，ところどころで，ユニゾン，3度や5度，6度のハモりをやって厚みをもたせたかったことにつきます。

歌をインストゥルメンタルにするときに，どうしても歌のパートが浮いてしまうなら，こういったハモりをもってくるのは，必須事項です。これはもうちょっとあとで詳しく説明しましょう。

フルートのまがいもののコーラスパートでは，オブリパートはどちらかというと高めの音でハモるので，感覚的にストリングをユニゾンさせました。オブリガードにストリングスはよくある手ですしね。サビでは特に。

その代わり，2本のストリングスで違うオブリを鳴らしてしまうと，ジャンル的に違ってしまう感じがあったので，楽譜1のストリングパートを多少いじって，オーボエに乗せかえたというところです。

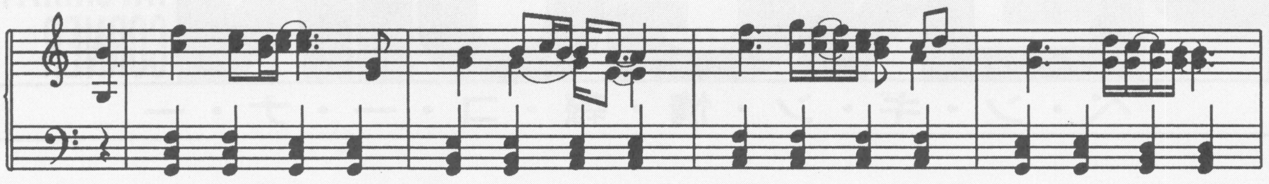
どうです？　これだけで十分違いが出たでしょ。歌のパートはこれだけやればほぼ完璧です。

§さらにアレンジ

さて，最後にもうひとつおまけで，ピアノソロにしてみました。

当然，右手メロディ，左手伴奏のピアノだけにしてしまうとメロディが薄くなるので，とある手段を使っています（特にシンセのピアノの音は，共鳴がないので薄っぺら。たとえCM-64でもね）。

では，楽譜3を参照。左手は手抜き（これは正真正銘の手抜き）で，原曲と同じく

楽譜3

コードの4分音符です。問題は右手のパートで，メロディが常に2音のインターバルで構成されているのがわかります。

これはたとえどんなジャンルだったとしても，ピアノソロへのアレンジのときには，必ずといっていいほど使われる常套手段です。練習用のピアノピース（楽譜）によく見られる，人気アニメソングのピアノソロ版などでも当然のように使われています。身近に落ちている「風の谷のナウシカ」のピアノソロ版でも，米米クラブの「君がいるだけで」のピアノソロ版でも，最近楽譜屋で見かけた「ドラゴンクエスト」のピアノソロ版でも使われています。

ポイントとしては，メロディの下にコード（三和音であったり，テンションを乗せたり）を置くこと。どうして上じゃなくて下かはわかりますよね。同じ音（ここではピアノ）が重なると，人間の耳はどちらかというと高い音を追ってしまうのです。だから，メロディはいちばん上にしておくのが無難なわけです。

単純に和音を重ねていくだけなら，コードに乗らない非和声音のときはどうするのかってことになりますよね。そんなときには3度下でハモりをやる，すなわち，3度下でも同じ種類の非和声音を使えばいいことになります。たとえばコードがCで，歌のメロディがEDEと続く場合，その下でもCBCと刺繍音（B）を入れてあげればいいのです。

また，もうひとつの手段として，強拍だけにしか和音を置かないという方法もあります。たとえばコードがFでメロディがFEGF〜と続くときは，下ではC（Fの構成音）を鳴らしっぱなしにすればいいのです。

これらのテクニックは，なにもピアノソロへのアレンジだけではなく，ちょっと音を加えたいときや，メロディパートが貧弱なときなどに，別の楽器でも使えます。ただし，もうちょっと本格的にアレンジするなら，ボイシングの知識が必要になってくるんです。いつか課題にしたいと思っているんですが，実は四声体の知識が必要となって，なかなかわかりやすい説明が思つかないんです。開離配置だとか。

リスト3

```
.comment 「忘れたくない友達のために」からサビの1部
.comment  作詞／作曲 紅呪威羅黄爵神
.comment  ピアノソロへのアレンジ  瀧 康史
.comment Roland CM64とそのコンパチ用 MML化 瀧 康史

(i)      /        初期化コマンド

(b1)     /        MIDI優先モード

(m1,2000) (aMIDI11,1)   / ピアノ右手その1
(m2,2000) (aMIDI12,2)   / ピアノ右手その2
(m3,2000) (aMIDI13,3)   / ピアノ左手

/ 本当は、ピアノパートぐらい1トラックでやりたい。
/ 贅沢をいわないから、2トラックでやりたい。(左右)
/ これもZMUSICのMMLが古典的なせいなんだろうな〜
/ しかし、これらの指定を1チャンネルにすると腐るのは、
/ きっとZMUSICのトラックミキシングの低性能のせいだろう。
/ ためしに、一度Mu1に持っていって、トラックミキシングして、
/ ならすと、きれいにうまくなるのである。
/ ちなみに、OPMファイルでピアノのものは、Mu-1に読ませて、
/ トラックミキシングして、MIDI化すれば1チャンネルで演奏
/ できます

(o70)

(t1) @5 @v120 @u090 L16 o4 g4 <f4e8de&e4.>g8   b4b8<c>b&ba8.&a4
(t2) @5 @v120 @u090 L16 o3 g4 <c4c8>b<c&c4.>e8 g4g4&ge8.&e4
(t3) @5 @v120 @u060 L04 o3 r4 'g<cf''g<cf''g<ce''g<ce''gb<e''gb<e''a<ce'
'a<ce'

(t1) <f4.gf&fed8c8d8 c4.dc&c>bb4.
(t2) <c4.cc&cc>b8a4  g4.gg&ggg4.
(t3)  'a<cf''a<cf''a<cf''a<cf' 'g<ce''g<ce' gb<d''gb<d'

(p)               /* 演奏命令
```

§ 最後に

さて，ここでインストゥルメンタルへのアレンジのコツを思いつくかぎり，箇条書きにしてみました。

●メロディを多少アレンジしてみる。歌詞に合わせるために細切れになった音をつなげてみたり，長く伸ばす音があったら，トリルや装飾音を入れてみる

●メロディにハモる音をユニゾンや3度，4度，5度，6度で重ねてみる

●（特に強拍では）メロディの下にコードなどを入れてみる。できたら三和音だけでなくテンションなどの特徴音も含めたい

●オブリガードを入れてみる

この4つを組み合わせて使い，さらに発展させることができれば，インストゥルメンタルへのアレンジもかなり楽に作業が進むでしょう。また，前述しましたが，ちょっと違ったら変だとかいう狭い聴き方をせず（もともとボーカルがないんだから，違うのは当たり前なんだし），おおらかな聴き方を身につけたいものですね。

＊　＊　＊

今回は応用編ということでしたが，役に立ちましたか。ところどころでコードの知識とか，いままでやってきた基礎知識も必要でしたが，大丈夫ですか？

まあ，勉強は必要に迫られてやるものですから，今回の記事を見て，“あ，これは前にやってたけどわからん”と思ったら，復習してくださいな。

いやはや，今回の原稿は書いてる側としては楽しめました。実際にやってみることが多いですからね〜。昔作った曲をデータにしたりアレンジしたりしてね。それとか，そこらに落ちてる楽譜を拾ってきて，ピアノにアレンジして弾いてみたり，毎回こんなのだったら楽しいのになあ（けっこう時間がかかるので，夏休みだからできたって話もなきにしもあらず）。

ところで，来月はなんにしましょう？まだ何も決まってません。“こういうのをやりたいんだけど，うまくいかないからなんかいい方法はないですか”とか，そんな相談がありましたら，お便りください。

アンケートハガキに書いてくれるよりかは私あての封書のほうがうれしいな〜，なんてぜいたくいったりして。アンケートハガキは編集部のもの，私に届いたお便りは私のもの。ではまた来月ね。

PENGUIN INFORMATION CORNER
ペ・ン・ギ・ン・情・報・コ・ー・ナ・ー

NEW PRODUCTS

電子タイトルプリンタ NS-TB1 コクヨ

NS-TB1

コクヨは電子タイトルプリンタ，タイトルブレーン「NS-TB1」を10月1日より発売する。

本機はSATZ(サッツ)ブランドの電子文具の第2弾で，タイトルラベルやインデックスラベルなどに，位置合わせを気にすることなく手軽に美しい活字でラベル作成を行える。

また,コクヨの128種類の既製タックラベルから目的に合わせて選ぶことができ，面倒な位置合わせをすることなく，レイアウトしてくれる自動レイアウト機能を搭載している。文字フォントとしてゴシック書体を標準装備。タックラベル以外にも，官製ハガキの住所印字やカセットインデックスカードのタイトルと曲目も，特殊編集機能により簡単に印字することができる。

そして，内蔵ファイルを使うことにより，使用頻度の高い文字を約750文字まで保存でき，さらにオプションのメモリカートリッジを使えば約2,000文字まで追加保存可能となる。

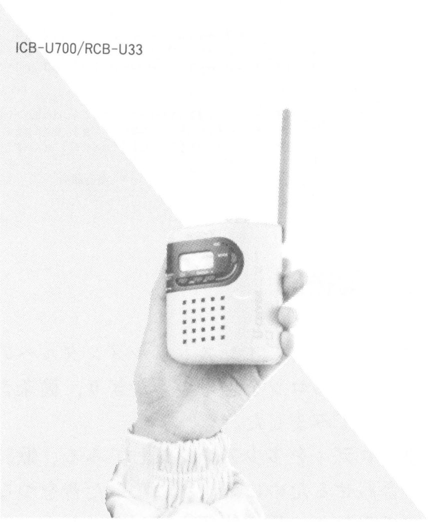
「NS-TB1」に対応するタックラベルは，従来の紙ラベルのほか，水や汚れに強い樹脂ラベルを新発売。4色のリボンカートリッジと合わせて効率的なファイリングができるようになっている。

価格は29,800円（税別）。

〈問い合わせ先〉

コクヨ㈱ ☏0120-032594

特定小電力トランシーバ ICB-U700/RCB-U33 ソニー


ICB-U700/RCB-U33

ソニーは，チャンネルの混み合うときにも円滑な交信ができるよう自動的に空きチャンネルを見つけるサーチ機能を搭載した，アームバンドスタイルの特定小電力トランシーバ“U-CEIVER＜ユーシーバー＞”「ICB-U700」(発売日：11月1日，ボディカラー：グレー，スポーツイエロー，スポーツグリーン）と，受信専用機「RCB-U33」（発売日：9月21日，ボディカラー：グレー）を発売する。

「ICB-U700」は免許，資格の不要な特定小電力トランシーバで，スキーゲレンデなどのチャンネルが混み合う場所での使用の際にも，スムーズに確実な交信ができるサーチ機能を装備したもの。

本機は，送受信側ともチャンネル番号を設定する必要がなく，交信の際に送信側のトランシーバが空きチャンネルを自動的に見つけ，相手を同じチャンネルに呼び出して交信を行うことができる。

このほかにも，同じグループ番号以外の交信が飛び込むのを防ぐ機能も装備し，従来どおりの9チャンネルで交信するシンプルな交信機能と合わせ，周囲の状況に応じて最適な交信方法を選ぶことができる。

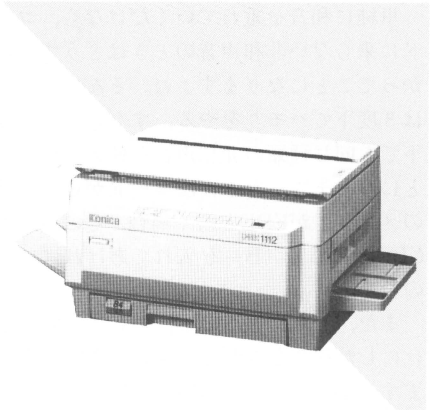
価格は，「ICB-U700」が28,000円(税別)，「RCB-U33」が10,800円（税別）となっている。

〈問い合わせ先〉

東京お客様ご相談センター☎03(3448)3311

関西お客様ご相談センター☎06(251)5111

コンパクトコピーマシン U-BIX1112 コニカ


U-BIX1112

コニカは，高画質，省スペースのコンパクト複写機「U-BIX1112」を発売した。

「U-BIX1112」は従来機より高画質化を図り，繰り返しコピーをとっても画像の再現性を向上させたほか，トナーリサイクル方式の採用など環境を重視したコンパクト普及機となっている。

細線やハーフトーンなども原稿に忠実に再現することができ，毎分12枚（A4）の複写機としては占有面積で最小クラスの省ス

ペースで設計がされている。

本体価格は278,000円(税別)となっている。

〈問い合わせ先〉

コニカ㈱　☎03(3349)5251

コードレス留守番電話機
CJ-A45
シャープ

CJ-A45

シャープは特定の子機，または親機にだけかけることができる「ダイレクトイン機能」搭載のコードレス留守番電話機「CJ-A45」を発売した。

本機の「ダイレクトイン機能」は，電話番号に続いて子機または親機の番号を押すと特定の電話機を直接呼び出すことができるもの。

そして，新電電などに加入していれば市外通話をかけたとき，自動的に各電話社の中から最も安い回線を選択する「α-LCR機能」も持っている。

子機は4台（合計6台）まで増設することができ，それぞれの子機は待受10日間，連続通話7時間が可能。さらに，「盗聴防止スクランブル機能」も搭載している。

本体価格は99,800円。また，別売のイヤホンマイク「DZ-E10」が3,800円（ともに税別）となっている。

〈問い合わせ先〉

シャープ㈱　☎03(3260)1161,06(621)1221

FORTHコンパイラ
MF68K (バージョンアップ)
マイクロフォース

マイクロフォースは，X68000用FORTHコンパイラ「MF68K」のバージョンアップ版を発売した。

「MF68K」は，前バージョンより高速化，コンパクトなオブジェクト生成を実現した本格的なFORTHコンパイラだ。

MF68K

Z80，8086，68000で統一された仕様を持ち，多くの制御文やインラインアセンブル機能などを備え，効率的なプログラミングを行えるようになっている。Human68kのDOSコール，IOCSコールを基本としたライブラリも装備しているため，X68000固有の機能を簡単に使うことができる。

コンパイラ本体のほかにアプリケーションプログラムとして，ソースレベルデバッガを標準装備。プログラムをROMライタに転送する「SNDROM」なども追加された。これらのアプリケーションは，すべてソースリストが付属している。

価格は19,800円(税別)，ユーザー登録している人には3,000円（製品一式，送料込み）でバージョンアップサービスを行っている。

〈問い合わせ先〉

マイクロフォース㈱　☎03(3756)1988

INFORMATION

第5回デスクトップ・ミュージック
「力作」コンテスト
ローランド/ローランドMCクラブ

ローランド，ローランドMCクラブでは，今年で5回目を迎えるパソコンを使った音楽コンテスト“第5回デスクトップ・ミュージック「力作」コンテスト”の作品募集を実施している。

募集期限は10月10日まで，参加は無料。応募部門は，オリジナル部門，アレンジ部門の2部門。総合グランプリには50万円相当，各部門グランプリには30万円相当のローランド製品を進呈，力作賞や敢闘賞，参加賞も予定している。なお，入賞作品の著作権はローランドに帰属する。

応募資格は，プロ，アマ，年齢，性別を問わず。グループでの応募も可。対象パソコンは，X68000シリーズ，Macintoshシリーズ,IBM ATバス/マイクロチャネルバス各機種，PC-9801シリーズ，FM TOWNSシリーズ(メディアは5インチおよび3.5インチフロッピーディスク)。

また，使用ソフトウェアは自作ソフトを含め，いっさいの制限はない。受賞作品の発表は1992年11月となっている。

〈応募先〉

〒101　東京都千代田区神田須田町2-11 ローランド(株)内　第5回デスクトップ・ミュージック「力作」コンテスト事務局

〈問い合わせ先〉

「力作コンテスト」事務局　☎03(3251)2831

Sapporo
Multimedia&CG'92
北海道コンピュータグラフィックス協会

11月24日～26日まで札幌パークホテルで「Sapporo Multimedia&CG'92」が開催される。本イベントでは，「デザイン」をテーマとして，CGやマルチメディアの技術を発表し，企業の情報交換，ならびに一般市民へのマルチメディアの啓蒙を図ることを目的としている。

プログラム内容は，CGシステム，マルチメディアに付随する周辺機器を展示した「情報機器展」，世界最新のCGを上映する「CGフィルムショウ」，そしてCG関連，マルチメディア関連論文の発表会や特別公演も行われる予定だ。

また，同イベントではCG作品コンクールも予定していて，アマチュアおよびプロフェッショナルまで広く作品を募集する。

募集部門は，コンピュータおよびデザインを職業としていないCG愛好家のための「アマチュア部門」と，コンピュータやデザインを職業としている人が参加できる「プロフェッショナル部門」で，それぞれ部門賞1点と優秀賞2点が用意されている。

応募規定は1991年12月以後にCGで制作した作品全般。2D，3Dなどの作品，また市販のソフトを使用して制作した作品でも参加は可。募集期限は11月17日（必着）。

〈問い合わせ先〉

Sapporo Multimedia&CG'92実行委員会事務局　☎011(232)3335

FILES Oh!X

このインデックスは，タイトル，注記──著者名，誌名，月号，ページで構成されています。季節も変わりましたね。運動会や遠足で活躍する人も，馬肥える秋の人も，読者のみなさんはやはり，秋の夜長はパソコン三昧かな？

参考文献

I/O　工学社
ASCII　アスキー
コンプティーク　角川書店
テクノポリス　徳間書店
POPCOM　小学館
マイコンBASIC Magazine　電波新聞社
My Computer Magazine　電波新聞社
LOGIN　アスキー

一般

▶Hardware Hot Press
各メーカーの新機種，周辺機器を紹介。計測技研のX68000用CD-ROMドライブ，オムロンのグレーイメージハンディスキャナ「HS20G-H」，横河・ヒューレット・パッカードの世界最小1.3インチハードディスク「HP Kittyhawk」，アップル社の電子手帳タイプコンピュータ「Newton」など。──編集部，POPCOM，9月号，26-30pp.

▶新鮮良品館
アウトドアでのコミュニケーションに便利なトランシーバー，シャープ「CB-T10」など，各メーカーの家電新製品を紹介。──編集部，POPCOM，9月号，126-127pp.

▶真夏のCG大作戦
CGイラストを描くときのノウハウや，各市販ツールの特徴や使い方，CGコンテストの常連さんのインタビューなど，これからCGイラストを描いてみたい人に役立つ情報を満載。──編集部，POPCOM，9月号，別冊付録

▶電脳通信
シャープのハンディHi 8ビデオカメラ「VL-HX1」や，各社のハイビジョン対応テレビなどを紹介。──編集部，コンプティーク，9月号，235-237pp.

▶THE NEWS FILE
アップルと東芝がマルチメディアで提携，CD-ROMプレイヤー開発で協力していく。巷の子供たちに大人気のバーコードバトラーの話題など，電子メディアや玩具のニュースを掲載。──編集部，LOGIN，15号，36-43pp.

▶特集　バラ色のハードディスク
いまやパソコンライフの必需品となったハードディスク。その基本原理や選び方，ゲームソフトをインストールして利用するときのノウハウなどを解説している。これからハードディスクをつないでみようというユーザーに役立つ記事。──編集部，LOGIN，15号，195-217pp.

▶ワープロ・パソコン通信新聞
NIFTY-Serveが提供する「スポニチ芸能情報」，PC-VANの「中古車情報」，ASCII NETの「F1情報」。EYE-NETの「アイドルチャンネル」など，大手ネットの最新サービスの紹介や，これからパソコン通信を始めたい人に役立つパソ通入門など。──山本まさこ，マイコンBASIC Magazine，9月号，90-93pp.

▶BASICプログラミング講座
「弾性衝突をゲームにしてみよう」。比較的簡単な物理法則を利用してゲームを作る講座。BASIC使用で教育教材としても使える。──東　幸太，マイコンBASIC Magazine，9月号，100-104pp.

▶どこでもいくぞ日本パソコン百景
最終回の今回は，月刊アスキー編集部を訪問。コンピュータの多さと机の乱雑さ，意味不明のはり紙が特徴とか。──フデヨシ＆カワラ，ASCII，9月号，160-161pp.

▶パソコン入門音楽編
MIDIを使ったコンピュータミュージック入門の特集。コンピュータミュージックの楽しみ方や機材選びについて。また，IBM PCの音楽環境の紹介も。──上野利幸・斎藤樹ほか，ASCII，9月号，197-212pp.

▶Digi-Ana Valley
日本マランツのAX1000を試す。リスニングルームの音響特性をDSPで解析し，位相や音響特性の歪みのないオーディオ再生を可能にする機械だ。世界のコンサートルームの音響特性を自宅に再現できるあたりは圧巻。──編集部，ASCII，9月号，245-252pp.

▶バカパパのモノを買い物
パソコン周辺グッズを取り上げるページ。9月号ではいままで取り上げそこねた製品をノンジャンルまとめて紹介する。──バカパパ，ASCII，9月号，314-315pp.

▶近代プログラマの夕べ
伝書鳩に関するおハナシ。第二次世界大戦中アメリカで行われていた「プロジェクト・ピジョン」とは？──ホーテンス・S・エンドウ，ASCII，9月号，330-331pp.

▶ラッキー！ハッピー！オッケー！
著作権にまつわる問題について弁護士に聞くページ。今回は「中古ソフトの売買，友人への貸与などは著作権に触れるか？」。──編集部，ASCII，9月号，336p.

▶パーソナルコンピュータこれまでの15年これからの15年
月刊マイコン創刊15周年記念企画。NECパーソナルコンピュータ販売推進本部長，富田克一氏へのインタビューや最近の国内外ニュースの紹介を通してこれからのコンピュータ界の展望を探る。──編集部，My Computer Magazine，9月号，60-81pp.

▶PC実験室
フロッピーディスクにはいろいろ注意書きがされているが，禁止されていることをやるとどうなるのか？MyCOM編集部があえて実験し，その結果を報告する。──編集部，My Computer Magazine，9月号，142-145pp.

▶MYCOM WATCHING
環境にやさしい発電方法として注目を浴びている風力発電。東北電力が津軽半島竜飛崎に建設した試験施設，竜飛ウインドパークの取り組みを紹介する。──菊地秀一，My Computer Magazine，9月号，206-209pp.

▶9600bpsモデム3機種を比較
今年の春の新製品から一段と買いやすくなった9600bpsのモデム。そのなかから主なもの3機種を取り上げ，ロードテストを行う。──高橋雄一，My Computer Magazine，9月号，225−229pp.

▶なんでもQ＆A
書院パソコンを使っていくうえでよく発生する疑問を7題まとめてシャープが回答する。──シャープ株式会社，My Computer Magazine，9月号，286-287pp.

▶ゲーム作りのノウハウ
ゲーム作りの好きな筆者を揃えておくる実践ゲームプログラミング講座。グラフィックや音楽の仕上げ方やWindowsプログラミングのノウハウなども公開。──STUDIO GAINI＆大澤文孝，I/O，9月号，34-65pp.

▶Programmer's Room
投稿プログラムや記事に対する読者の意見・感想を取り上げるページ。線分描写プログラムの高速化の方法は？　などのトピックスを収録。──ななふし，I/O，9月号，85-87pp.

▶大道芸ロボット大会
7月5日に機械産業記念館で行われた「大道芸ロボット大会」の模様をリポート。ケン玉をする，コップを叩いて曲を演奏するなど，おかしなロボットたちが出展されていた。──尾崎晶一，I/O，9月号，103-105pp.

▶スーパーコンピューティング入門
従来からある数学の難題を，コンピュータの力を借りて解決していく。今回はN-QUEEN問題を取り上げ，まずその内容を解説する。──林智雄，I/O，9月号，140-142pp.

MZシリーズ

MZ-1500(BASIC MZ-5Z001)

▶移植版ピラミッド・パズル
トランプを使ったパズルゲーム。総和が27になるようにカードを取っていく。──白井建夫，マイコンBASIC Magazine，9月号，120-121pp.

MZ-2500 (M25-BASIC)

▶重力地獄
ヤブ君を操って鉱石をすべて取ればＯＫ！　重力の方向が変わるので注意。重力の変化を使ったパズルゲーム。──謎のパズル大好きおじさん，マイコンBASIC Magazine，9月号，122-124pp.

X1/turbo/Z

X1シリーズ

▶暗闇
暗闇の迷宮から鍵を3つ探し出し，扉から脱出。暗闇の恐怖に耐えられるか！　──有蹄類，マイコンBASIC Magazine，9月号，151-152pp.

▶JANKEN WARS2
ジャンケンで勝って勝負に負ける⁉　上から2人で手を落としていって横か斜めに5つ並べると勝ち。──さぶ，マイコンBASIC Magazine，9月号，153-154pp.

X1turboシリーズ

▶大連
麻雀パイを操作して面子を4組つくる。コラムス風ア

クション・パズル麻雀。――BAKE青柳，マイコンBASIC Magazine，９月号，155-156pp.

X68000

▶NEW SOFT
開発中ゲームソフト紹介。かわいいキャラが大活躍「エトワールプリンセス」。――編集部，LOGIN，15号，25p.

▶X68000新聞
アメリカ生まれの源平シミュレーションゲーム「ライジングサン」，「太閤立志伝」，「リーディングカンパニー」，「MATIER」，ポピュラスシリーズの新作「ポピュラスII」。新着ゲームとツールを紹介。――編集部，LOGIN，15号，226-229pp.

▶GAMING WORLD
熱い３人組が大暴れ！　格闘アクションゲーム「ファイナルファイト」の攻略と，「ライジングサン」，手に汗握るカーチェイス「チェイスH.Q.」などを紹介。――編集部，テクノポリス，９月号，17-30pp.

▶レモンちっくWORLD
新着美少女ゲーム大紹介。「テセラ」，「ARMIST」，「新宿物語」，「DRAGON PINK」，「PLERIA」ほか。――編集部，テクノポリス，９月号，85-99pp.

▶Software Hot Press
開発中のゲームソフトを紹介。「ライジングサン」と，ズームの新作カーレースゲーム「オーバーテイク」。――編集部，POPCOM，９月号，21-22pp.

▶SOFT EXPRESS
各機種の新着ゲームを紹介。「ライジングサン」，「チェイスH.Q.」。機種別の新作ソフトのリストあり。――編集部，コンプティーク，９月号，59-62pp.

▶NEW SOFT
「ポピュラスII」，「ライジングサン」，立体ヘックスで戦うシミュレーション「シュート レンジ」などの新作ゲームソフトを紹介。――編集部，LOGIN，16・17号，24-39pp.

▶最新ゲーム徹底解剖!!
カプコンのアーケードゲームからの移植，待望の格闘ゲーム「ファイナルファイト」を攻略。ステージ３まで。――編集部，LOGIN，16・17号，194-195pp.

▶X68000新聞
ナディアファン待望の「ふしぎの海のナディア」が９月下旬ついに発売。ズームの新作「オーバーテイク」。SX-WINDOW開発キットもリリース間近。――編集部，LOGIN，16・17号，288-291pp.

▶HOT！ INFORMATION
９月頃にリリース予定のX68000用SX-WINDOW対応ソフト「Communication SX-68K」，「SOUND SX-68K」，「SX-WINDOW開発支援ツール」を紹介。――編集部，マイコンBASIC Magazine，９月号，97p.

▶DIVE
お猿のモン吉君を操り，空中ブランコから移動するトランポリンへと飛ぶ空中ブランコアクション。――梅津毅，マイコンBASIC Magazine，９月号，157-158pp.

▶Castle build
バラバラになった城をロボットで元に戻してください。城作りパズルゲーム。――まいまいかぶり，マイコンBASIC Magazine，９月号，159-160pp.

▶Belldeer Wind
セガのゲームミュージックプログラム。要NAGDRV。――岡田大輔，マイコンBASIC Magazine，９月号，174-176pp.

▶今月の注目パソコンゲーム
カプコンのX68000参入第１弾「ファイナルファイト」を取り上げる。――いたばし，マイコンBASIC Magazine，９月号，225p.

▶AV STRASSE
X68000のウィンドウプログラミングを可能にする開発ツール群「SX-WINDOW開発キット」，AS.X上位互換のハイスピードアセンブラHAS.Xなどを紹介する。HAS.Xとリンカhlk.xは付録のディスクに収録されている。――編集部，ASCII，９月号，269-274pp.

▶FREE SOFTWARE INDEX
大手BBSにアップロードされたソフトウェアのなかから編集部が選んだものを紹介する。X68000用シューティングゲームIKR.X，SRAMユーティリティSRAMCLR.Xなど。――編集部，ASCII，９月号，349-355pp.

▶なんでもＱ＆Ａ
ヒストリデバイスドライバのHISTORY.HISの機能は何か？　またBUSINESS PRO-68K Popularで自分の作成した外字を使う方法は？　などの質問に答える。――シャープ株式会社AVCシステム事業推進室，My Computer Magazine，９月号，284-285pp.

▶SLG Laboratory
光栄から発売されたシミュレーションゲーム「三國志III」について，テストプレイの模様を歴史小説風に仕立てて紹介する。――猪野清秀，My Computer Magazine，９月号，312-315pp.

ポケコン

PC-E500

▶空手
「ストリートファイターII」を目指した格闘アクションゲーム。――打越丈晴，マイコンBASIC Magazine，９月号，162-165pp.

PC-E550

▶誌上公開質問状
カナ文字フォントの書き換えをすることはできるか？――BATIAN，マイコンBASIC Magazine，９月号，178p.

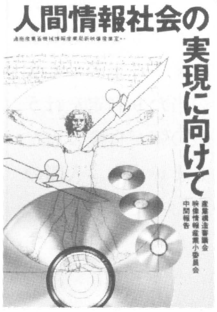
新刊書案内

パソコン用語
迷解辞典
松下典望著
秀和システム
トレーディング刊
☎03(3470)4941
A5判　265ページ
1,500円(税込)

パソコン用語辞典が役に立たないのは有名な話で，知っている言葉に対しては肝心な記述がなく，知らない言葉を引くと何が書いてあるかわからない。そもそもコンピュータ用語なるもの，語源辞典が必要なほど意味が錯綜，伝播，曖昧，浸透と拡散しており，同じ言葉でも，俗語的な用法，ハードウェア回路での用法，アーキテクチャにおける用法，システム内における用法，ユーザーとコンピュータの関係における用法などで示すものが微妙に違ったりする。特に，コンピュータにとって重要な言葉，たとえばインタフェイス，アクセス，環境，システム，メディアなどで顕著だ。

本書はそうした，特に重要で特に曖昧な単語12個を取り上げ，曖昧な部分を徹底的に追究し，既存の用語辞典やコンピュータ関係の書籍，果ては国語辞典や英語辞典まで俎上にのせ，その頓珍漢さを言及しつつ，その本質を浮かび上がらせようとする。本質が浮かび上がればいいが，著者の目的はおそらくそこではない。いかに言葉がいい加減に使われているか，いかにシステムという言葉が便利なものかを暴くことによって，問題を浮き彫りにすることだ。そのため，引用文献は多岐に渡り（そもそも引用とそれに対するコメントが中心だ)，非常に面白く読める。

かなり強引な展開，揚げ足を取っているような部分（一見その文がおかしくても，文脈によってそれが適切になる場合もある)，素人が読んだらかえって混乱するだけ，文体がくだけているようでくだけていない，勢いはあるが論理的な整合性はいまひとつ，といった独善的な傾向はあるものの，パソコン用語の曖昧さや使われ方のいい加減さに焦点を当てた本はなかなかないため，本書を読んでその曖昧さや，曖昧さが蔓延した原因を知ることができれば十分に有益だといえる。自らの経験と見栄だけで言葉を垂れ流している多くの人々にぜひとも読ませたい。あ，私のことか。　（K）



人間情報社会の実現に向けて
通商産業省機械情報産業局新映像産業室編
財団法人
通商産業調査会刊
☎03(3535)4881
A5判　88ページ
1,400円(税込)

本書は，通商産業大臣の諮問機関である，産業構造審議会映像情報産業小委員会の報告書を中心として編集されたものである。映像情報について，その本質的特徴や社会との関わりの分析にはじまり，ソフトの制作，流通，技術，社会システムの在り方など幅広い分野における環境整備の必要性が提言されている。

メディア技術や電子映像技術の発展や，入力・画像処理技術の動向などの具体的データ，あるいは概念図などについては，参考資料として図表が添えられている。また，関連用語の解説，海外調査報告概要なども併せて収録されている。

オールザット
ウルトラ科学まんが
①
米田　裕著
アスキー出版局刊
☎03(3486)1977
A5変形判　184ページ
1,200円(税込)

LOGIN誌上で，1985年から現在まで連載中の「オールザットウルトラ科学」。そのイラストのまんがだけを集めたのが本書である。

記事の本文のほうはサイエンスライターの鹿野司氏による科学うんちくエッセーだが，それを読んでいない人でも，これだけで十分楽しめる１冊だ。８年にわたる長期連載ということで，その時その時の話題や流行が垣間見えて，時代の流れが感じられるのもまた，面白さのひとつといえるだろう。

なお，本書のほとんどはDTP技術により作られたとのことである。

QUESTION and ANSWER

Q **X68000でシューティングゲームの当たり判定で悩んでいます。いま16×16ドット同士のキャラの当たり判定をするとき，**

```
if abs(x1−x2)<16 and abs(y1−y2)<16 then 当たり
```

ということをやると，たとえば自分の弾3発と敵10機のとき，3×10＝30回も判定しなければなりません。また，

```
dim int vram(15, 15)
```

として，敵の位置を，

```
vram((x1−8)/16, (y1−8)/16)=敵番号
```

というふうにすると，自機の弾の3つ分の3回で判定がすむけれど，正確な判定ができません。網目を細かくする。たとえば，

```
dim int vram(31, 31)
```

とデータを書き込む量が多くなってしまいます。なにか速くて正確な当たり判定はないでしょうか。

東京都　小川　淳

Q **ゲームソフトなどのキャラ（スプライト）同士の衝突判定の方法を教えてください。**

青森県　内山　勇児

A X68000のスプライトの大きさは16×16ドットです。したがって2つのスプライトが重なっているかどうかの判定は，それぞれのスプライトを表示しているX座標，Y座標の差の絶対値がともに16以下であるかどうかを調べればすみます。差がともに16以下なら重なっていると判定できます。スプライトAが(x1, y1)，スプライトBが(x2, y2)に表示されているとき，X-BASICでの当たり判定は小川さんの質問の中にもあるように，

```
if abs(x1−x2)<16 and abs(y1−y2)<16 then 当たり
```

となります。上式ではスプライトが1ドットでも重なると，当たり処理になってしまいます。スプライトの重なりにある程度余裕を持たせたいときは，16をもう少し小さい値（14とか）に変更すればいいことはわかりますね。

それから小川さんの質問にある「速くて正確な当たり判定」についてですが，私にはわかりません。たまたま居合わせたスタッフにも聞いてみたのですが，これといった解決法は聞くことができませんでした。

弾3個と敵10機で必要な30回の当たり判定を省略することはできません。特別な条件の下に簡略化できる場合もありますが，そのための付加処理が必要になりますので，コーディング次第では素直にやったほうが軽くなることも十分考えられます。粗い網目でだいたいの判定を行って細かい判定を行う方法なども同様です。

マシン語プログラムであれば動作速度が速いので，キャラクタの移動量なども十分小さく，当たり判定をメインループ2回に1回の割で行うこともできるでしょう。しかし，あらゆる場合に通用するとは限りません。

というわけで，私の勉強不足のために当たり判定を高速化する話はできません。その代わり正確な（というより自然な）当たり判定を行うための話をしたいと思います。

自機が戦闘機のグラディウスのような横スクロールシューティングゲームを作るとしましょう。自機はスプライトを2つ横に並べて32×16ドットの大きさとし，敵の弾は1個のスプライトで表示することにします。図1のようにスプライトに定義しました。普通にスプライトの当たり判定を行うと図2のような場合も敵の弾が自機に当たっていると判定されてしまいます。人間の目から見ると「いまの当たってないように見えたけど」となってしまい，不自然です。

そこで図3のようにスプライトに長方形の判定領域を設定します。そして判定領域に対して当たり判定をすると，見た目に自然な結果が得られるはずです。図3の判定領域の場合，敵の弾が水平に移動して自機の主翼や尾翼をかすめても，そこに判定領域がないので弾に当たったとはみなされません。俗にいうスリ抜けです。ヒットチェックのない部分は業務用のゲームでもありますよね。もし不自然な当たり判定があったら，判定領域の大きさを調整すればいいのです。判定領域を小さめにとっておいたほうがうまくいくようです。モノの本によれば市販ゲームのスプライトの当たり判定も，いま説明したような判定領域を設定しているものが多いんだそうです。

判定領域にこだわらずに，X-BASICでスプライト同士の重なりあいを高速に調べたいというのなら，アセンブラでスプライトの当たり判定ルーチンを自作して外部関数に組み込むという手もあります。実をいうと，かなり以前にスプライトの重なりを調べる外部関数が本誌に掲載されています。掲載号は1990年4月号で，（て）のショートプロぱーてぃに掲載されたsp_chk()です。またsp_chk()は1990年6月号の「創刊8周年記念PRO-68K」にも収録されています。

Q **市販のゲームの中を見ると，マップデータやスプライトデータらしきもの（拡張子が.DATや.CHR）が入っていますよね。ゲーム中に，たとえば町から外に出たとき，外のマップや敵キャラクタを読んで表示させてますよね。プログラム上でそのようなマップデータファイルなどを読み込んで，表示設定させる手順を教えてください。またマップデータや敵スプライトデータを読んで設定表示すると主人公のスプライトデータが消え**

図1

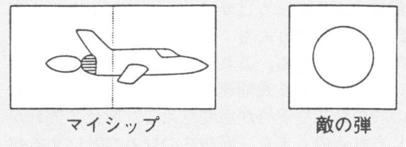
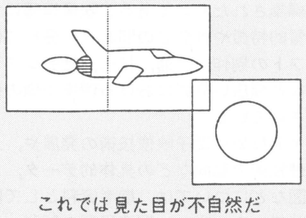
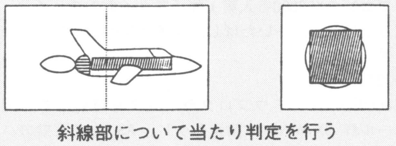


図2



これでは見た目が不自然だ

図3



斜線部について当たり判定を行う

てしまいますよね？　消える前にそのデータ分のメモリを確保して，スプライトデータを格納しておくのでしょうか？　あとスプライトとBGデータエリアの障害物の当たり判定ルーチンの作り方を教えてください。

東京都　小松隆夫

X68000の256×256モードでBGを2面使うものとして説明しましょう。BGを2面使うとき，スプライトパターンを128個定義することができます。128個で町の背景，敵キャラなどすべてのキャラクタを定義できるなら，ゲーム途中にスプライトデータを読み込む必要はありませんね。

もしも128個のパターンで足りないのなら，場面に応じてパターンデータを取っ替え引っ替え定義するよりしかたありません。質問の例ですと，町を表示するためのパターン，町の外の風景を表示するためのパターンをあらかじめディスクに（余裕があればメインメモリ上に）用意しておいて，必要に応じてパターンを定義することになるでしょう。

小松さんはアセンブラを使っているようですから，以下，マシン語レベルで解説します。普通パターンの定義はIOCSコールSP_DEFCGを使うと思います。ここでもIOCSコールでパターンを定義してもいいのですが，大量のPCGデータを速く書き換えたいのであれば，IOCSコールを使うのは得策ではありません。

スプライトやBGに定義するパターンデータはどこに格納されているのでしょうか？　答えは……“PCGエリア”と呼ばれるアドレス領域に格納されています。PCGエリアはスーパーバイザ領域にあるので，ふだんはアクセスすることができません。

ここではIOCSコールを使わずに直接PCGエリアにデータを格納してパターンを定義することにしましょう。問題はPCGのパターンコードに対応するPCGエリアのアドレス算出方法です。BG2面のときにPCGエリアの配置されるアドレスは，$EB8000～$EBBFFFです（BG1面なら，$EB8000～$EBDFFF，BGを使わなければ$EB8000～$EBFFFF)。

また，256×256モードでのBGの大きさは8×8ドットです。このときひとつのBGパターンデータは32バイトのメモリを必要とします。ですから，

PCG_ADRS＝$EB8000＋パターン番号×32

で求められます。したがってパターンコード0～63まで定義すると，PC_GADRSから64×32＝2048バイトに定義したパターンデータを転送すればいいことがわかります。

PCGエリアに転送するパターンデータファイルを作成するには，デバッガを使って現在のPCGエリアに格納されているデータをファイル出力して行うことにします。スプライトエディタなどを使って，パターンコード0～63に町を表示するためのPCGを定義したとしましょう。データはPCGエリア$EB8000～$EB87FFに置かれます。コマンドラインから，

```
A>SCREEN 1
A>DB
X68k Debugger Version X.XX ……
-WA:¥MACHI.DAT, EB8000, EB87FF
```

とします。これでパターンコード0～63に定義されているパターンデータをMACHI.DATにファイル出力することができます。最初にSCREEN 1をしているのは，スプライトとBGを使うことのできる画面モードに変更するためです。そうでないとPCGエリアをアクセスしてもバスエラーが発生するからです。

同様に町の外の風景などBGに使うPCGデータもファイル出力してください。

実際のプログラムがどういうものかわかりませんが，スプライト，BG画面をフェードアウトしているあいだにPCGデータを書き換える方法が，いちばん簡単だと思います。町から町の外に移るときに画面をフェードアウトさせ，そのあいだにDOSコールを使ってファイルをPCGエリアに転送すればいいのです。

それからマップデータファイルを読み込んで主人公のPCGデータを消してしまっては，主人公の表示ができなくなってしまいます。どの場面でも共通して使うPCGデータがあると思います（木，岩，主人公とか)。たとえば共通のパターンデータをSPコード100～127にまとめて定義しておくようにして，それ以前のSPコード0～99のパターンのみ変更するようにしてください。

BGとスプライト画面の衝突判定について簡単に説明します。表示画面が256×256のとき，スプライトが16×16でBGが8×8なので，スプライトと重なるBGの数は4～9個です。実際にはキャラクタの移動方向などによって場合分けすれば判定個数は軽減することもできます。

BGパターンコードの0～31のあいだに障害物（スリ抜けることができない）パターンを定義し，32から63のあいだにスリ抜けることのできるパターンを定義，というように範囲を決めておくと，衝突チェックが楽になります。

抽象的な話が続いたので具体例を出しましょう。衝突判定するスプライトの左上が(X，Y)にあるとき，それと重なっているBG領域はX MOD 8＝0，Y MOD 8＝0 (X/8, Y/8)－(X/8+1, Y/8+1)の範囲ですね。X MOD 8≠0，Y MOD 8＝0なら(X/8，Y/8)－(X/8+2，Y/8+1)，X MOD 8＝0，Y MOD 8≠0なら(X/8，Y/8)－(X/8+1，Y/8+2)，X　MOD 8≠0，Y MOD 8≠0なら(X/8，Y/8)－(X/8+2，Y/8+2)が重なっている範囲になります。

BG座標に対応するBGデータエリアは，

BG_ADRS＝$EBC000＋Y×128＋X×2

です（BG0の場合。BG1なら$EBC000を$EBE000にする)。BGデータエリアはひとつの座標に1ワードとります。下位バイトがPCGパターンです。つまりBG_ADRS＋1にBGに表示されているPCGパターンが格納されています。これを調べて0～31なら障害物に衝突，それ以外ならそのまま，とすればいいと思います。　（影山裕昭）

質問にお答えします

日ごろ疑問に思っていること，どんなことでも結構です。どんどんお便りください。難問，奇問，編集室が総力を挙げてお答えいたします。ただし，お寄せいただいているものの中には，マニュアルを読めばすぐに回答が得られるようなものも多々あります。最低限，マニュアルは熟読しておきましょう。質問はなるべく具体的に機種名，システム構成，必要なら図も入れてこと細かに書いてください。また，返信用切手同封の質問をよく受けますが，原則として，質問には本誌上でお答えすることになっていますのでご了承ください。なお，質問の内容について，直接問い合わせることもありますので，電話番号も明記してくださいね。

宛先：〒108 東京都港区高輪2-19-13
NS高輪ビル
ソフトバンク株式会社出版部
「Oh! X質問箱」係

STUDIO X ON AIR

FROM READERS TO THE EDITOR

夏休みも花火も西瓜も海水浴も盆踊りも蟬の声も，みんな終わっちゃったけれど，運動会や紅葉狩りや栗拾いや鈴虫やあかとんぼや，季節が変わると新しい楽しみがありますね。グラフィックツールを駆使して，「芸術の秋」とかね。

◆1.5年前くらいから読ませていただいている者ですが，ハガキを出すのは初めてです。実は「正しい花瓶の落とし方」にかんどーして出しました。自分の知識を総動員し，実験を繰り返し，みごとに結果を出してしまう，科学者の鑑のような方ですね。柴田さんという方は。しかも文系とは！　僕も反省して，おべんきょーすることにします(ほんとか？)。川井　敦(21)東京都

文系とか理系とかではなく，好奇心を持って追究していくことが大切なのでしょう。

◆プログラムを作る手順はやはりフローチャートが先なのでしょうか？　学校の演習などでフローチャートが必要なことがありますが，どうしても完成したプログラムからフローチャートを書いてしまいます。皆さんはどうでしょうか？　笹川　純一(21)埼玉県

◆8月号の特集の中の「プログラムの流れをつかもう」はよかった。僕自身も約10年前，あのファミリーベーシックでこういう考えをし，頭をかかえたものでした。しかし，こういうことを小学生のときにやっていたので，高校の情報処理のときは，フローチャートを書く前にプログラム(BASIC)を作って，よく，先生に注意されていました。高校で習うBASICごときではフローチャートはいらないと思うのは僕だけでしょうか。　野崎　国彦(18)岡山県

確かにフローチャートを書くのは面倒なときもありますが，考えを整理するためにフローチャートを書く習慣をふだんからつけておいたほうが，大きなプログラムを作るときのためにはいいかもしれませんよ。

◆私にとって，プログラムを作るのに大変なのは打ち込みで，なんとかいい方法がないかと，いつも思います。　長沼　正(28)東京都

原稿を書くのに大変なのも打ち込みで……。頭に電極でもつないで，考えたことがそのまま入力されるとかいうのはどうでしょう？　でも，ちょっと怖い気もする。

◆プログラミングだけでなく，大きなプログラムみたいなものを作るときの理論なども学校で学んでいます。小さなツールくらいのプログラミングはよくするんですが，大規模なものになると，さすがにちょっと……と思ってしまいます。やっぱり，要は精神力の問題なんでしょうね。　小川　靖浩(21)東京都

精神力と体力！　プログラミングも大きなものだと，「力仕事」的な部分もけっこうありますからね。

◆丹さんの「作り散らかせます」はよかった。常々私が思っていたことを代弁してくれた。X Toolkitを知る身としては，ウィンドウプログラミングをオブジェクト指向的に行うことがごく自然であり，自分で何からなにまでバキバキやってしまうのは美しくない。早くシャープからSX-WINDOW用プログラミング開発ツール(キット)がSX Toolkitとして発売されてほしい，と思っていた。しかし，丹さんはこのSX Toolkitをプロトタイプのみならず，きちんと完成させるつもりなのか，あるいは，あくまでアプローチにすぎないのかが少し気になる。だってこれはシャープが本来やるべき仕事では……？。

寺田　泰(23)北海道

◆SX Toolkitは，面白い構想だと思います。なにかとハードルの高そうなウィンドウシステムのアプリのプログラミングですが，乗りこえる有効な手段のひとつはやはりオブジェクト指向でしょう。g++も手に入れたし(Cマガジンに感謝)，C++の世界に触れてみようかと考えています。　佐藤　充浩(21)長崎県

オブジェクト指向言語の今後が期待されますね。

◆「C++」の正しい読み方をご存じですか？　私は「シー・インクリメント」だと思っていたのですが，某雑誌に「シー・プラスプラス」と書いてあったので，少し疑問に思ったもので。

山内　啓史(23)福井県

え，あれって「シー・プラプラ」じゃなかったの？　なんて。パソコン辞典には載っていないようですが(製品名ですからね)，メーカーさんによると正しくは「シー・プラスプラス」とのことです。

◆8月号の表紙を見て，NTTのCM(わ～になって……)を思い出してしまった……(笑)。

伴　哲也(21)京都府

◆8月号の表紙の天使みたいなのは目つきが悪い。　小沼　健太郎(15)千葉県

◆8月号の表紙は少し恥ずかしい。

西岡山　誠人(20)奈良県

8月号の表紙については，このほかにもいくつかハガキをいただきました。人気者になって，天使も喜んでいるでしょう。

◆8月号の「響子in CGわ～るど」を読んで，とっても感動しました。「羽がもげた」というひと言がとっても寂しさを感じさせます。僕の羽は，ついていても動いていない。今度，少しでいいから動かしてみようと思います。

桒野　拓也(19)東京都

◆「響子in CGわ～るど」は，何か，いま現在の自分に大きな衝撃を受けました。私は，今年の3月に考えるところがあって会社をやめたのですが，この「CGわ～るど」の記事は，そんな自分に，いったい何がいちばん好きなことで，何がいちばんやりたいことなのだろうと，最近少しずつ堕落してきた私の心に弾丸を打ち込まれたようでした。そして，いつかは飛んでやる，と思わされました。　浅場　誠志(20)神奈川県

「CGわ～るど」にも，いくつかのハガキを

▲横井　賢一　富山県

OVER TAKEのイラスト。可愛い女の子もよいけど，後ろのエンジンの細かさもなかなか。こんどはマシンを描くのにも挑戦してみては？

▲鈴木　貴久　静岡県

ジュゴンとマナティーの違いは私も知りませんでした。でもどちらにしても，やっぱり私は美人の人魚のほうが好きだなあ。

いただきました。自分の羽で飛び続けるのは，つらくて苦しいことですが，飛べなくなってしまうのは，もっとつらい気がします。少しずつでもいいから，飛び続けたいですね。たとえ休みながらでも，ね。なくした羽だって，「飛びたい」って思い続ければ，きっとまた生えてくるでしょう。

◆メガネ屋でX68000が活躍してるそうですが，インテリアのコーディネイトショップみたいなところでも，いろいろ使われているようですよ。家具や照明器具の取り込み画像と，部屋のCGを組み合わせてお客さんにイメージを作ってもらうやつです。かっこいいよなあ，と思いますよね。よっX68000。　高橋　政秀(19)東京都

◆7月号のメガネ屋のX68000の発見者は私です。また，日立市内の整体医院にもX68000があります(表から，初代の裏が見える)。さらに，昔，水戸駅ビルでX68000が中古車販売のデモをやっていた。そういう私は浪人で，キーボードとマウスを取られてしまいました。

関口　道(18)茨城県

社会で活躍中のX68000。

◆仕事に使おう，CGしよう……と思って買ったXVIだったが，業務用には，モニタ付き8万円で譲り受けたPC CLUBを使い，XVIはゲームマシンとなってしまった。しかし，DōGAも手に入ったし，一発トライしてみるぞぉ！　早く68030(または40)を積んだX68000が出ないかなぁ。

坂倉　豊年(27)神奈川県

◆1カ月遅れでDōGA CGAに力を入れだした。マニュアルがないので，全体のイメージはつかめても，細かな使い方が「？」。80Mバイトのハードディスクを整理して，お試しディスク内のデータの解析を始め，方眼紙に図面を描き，データをエディタで入力して，CADで表示して満足している。データ入力はダイナパースに慣れているため，データコンバータでも作ろうと思う。

高木　宣博(22)大阪府

先月号にひき続き，ますます好評のDōGA CGAシステムです。使い方も人それぞれのようですが，皆さん，がんばってくださいね。

◆2048×2048ドットのフルカラー画像エディタMATIERが39,800円。思わず衝動買いしてしまいそうです。ついでにフルカラーフレームバッファがお安く出ればいいのですが……(某98用では定価59,800円のボードがあるそうです)。MIRAGE Systemも発売になったし，DōGAもがんばっているし，最近になってX68000第2世代CG時代が訪れたような気がします。わくわく。

池内　義直(25)栃木県

◆MIRAGE SystemにMATIERと，ここにきてようやくユーザーが本当に欲しかったツール，プロ使用に耐えるようなグラフィックツールが出てきました。いままでZ's STAFFにまかせきりだったPAINT機能や，512×512ドット，65536色(32768色)に限定された世界から，MATIERによって解き放たれました。X68000も，「なんでもそこそこできるパーソナルユース(のみ)のパソコンから，「やろうと思えば，プロフェッショナルな使用に耐えるパーソナルワークステーション」へとステップアップしなければなりません。そのためには何が必要であるのかを，ソフト，ハードの両面から誌上で述べていただきたいと思います。　細見　格(24)大阪府

▲新谷　寿成　東京都

さて，質問です。この女の子の左手はどうなっているでしょうか。ぐんぐん伸びていって，敵にからみついているかも……。

▲尾形　雅治　広島県

うん。S端子，欲しいよねえ。狭い我が部屋でもディスプレイテレビは大活躍。X68000や，某社のパソコンや，某々ゲーム機や，ビデオや……。

高機能で低価格のツールの出現で，X68000はゲームマシンではなく，ゲームを含めた強力なパーソナルホビーマシンなんだ，という認識が，ユーザー以外の人にもひろがるような気がします。これからが楽しみですね。

◆7月号のハガキに「グラフィックプロセッサ」というべきソフトが欲しい，などと書いたら，8月にはこれである。MATIERである。この前，Z's STAFF買ったのに。読んでいるのか，読まれているのか。　田中　幸夫(23)岡山県

「先見の明がある」と思っておきましょう。

◆プログラムを作成する場合，作るという「意志」とともに，作成ツールが必要となります。このツールが簡単で便利であれば，プログラム作成の苦労は少なくなり，途中であきらめることも少なくなるでしょう。SX-WINDOWに限らず，GUI上のプログラミングは苦労が多そうですが，今回の開発キットで，どれくらい楽ができるか注目したいです。　広井　誠(30)新潟県

◆やっとSX-WINDOW開発キットが出るのか。これで，SX-WINDOWのアプリケーションもどんどん作れ……ん？　メインメモリ4Mバイト以上？うーむ。　坂井　純一(22)茨城県

メインメモリ4Mバイトはちょっとつらいですが，簡単で便利なツールが出るのは，楽しみですね。

◆先日，電子手帳PA-X1を購入しました。メモリが32Kバイトしかないので心配しましたが，これがなかなかいっぱいになりません。それどころか，半分にも達しないのです。なんだかX68000って贅沢ですね。　谷　聡雄(19)茨城県

うーん，確かになんとなく贅沢でリッチな気分。でも，パソコンを使っていると，メモリとかハードディスクとか，どんどん贅沢な欲求が出てきますよね。たとえ，バブルがはじけても，もう，元の暮らしには戻れない!?

◆それにしても，下田達也さんはすごい。X68000を4台も使っているなんて……。でも，本格的にCG(特にレイトレ)をやろうと思ったら，必然的に複数台を使わなければならなくなるのでしょうね。　石川　正史(28)北海道

さらに贅沢なお話が出てきてしまいましたが……。まあ，ハード環境の不足分は「ハート」で補おう！　と，思っておこう。

◆「作りたい」と思っても，完成はちっとも目前にありません。それは，野望が大きすぎるからです。友人にこのことを話すと，根性が足りんのひと言。うーむ，確かにそのとおりなのです。そこで次の付録ディスクに以下のものを付けてくれると幸いです。

1)　X68000ができることを説明して試すディスク
2)　高度テク演出ディスク
3)　バグの対処&スピード強化ディスク

平井　章(20)岡山県

その「野望」と「根性」で，まず1～3のディスクを作って，Oh!Xに投稿してみてくださいな。

◆4Mバイトでもメモリが足りないと思う今日この頃。130Mバイトのハードディスクも残り20Mバイトくらいになり，なんか狭くなってきたと感じる。スロット2つもすでにSCSIとメモリボードに占領され，買いたいソフトもなかなか見つからないが，そんなことより，就職先が見つからないのが一番の問題のような気がする。

藤田　真史(24)北海道

今後の豊かなパソコンライフのためにも，就職活動がんばってください。

◆1989年3月，村田氏のマシン語プログラミングの連載が始まった号から購読しています。ところで，マシンの封印はやはり8月前にしたほうがよいのでしょうか。ああ，それでも半年しかない。2浪はいやだ……。

猪狩　友則(19)千葉県

封印してしまっても，禁断症状に耐えられなくなるかもしれません。「X68000は1日1時間まで」とか決めたらどうでしょう？　小学生のファミコンみたいだけど。

◆ある日，テレビを見ながら枝豆を食べていました。するとなにやら舌ざわりのよくないものがあったので，手に出してみました。よく煮えた小さな緑色のイモムシでした。ああ，夏のにがい思い出。　　　　　中島　民哉(21)埼玉県

◆この前，自販機でコーラ(びんのやつ)を買ったら，なんと，中身がスミでした。お店の人の話だと，サンプルを間違って入れてしまったということなのですが，それにしてもびっくりしました。　　　　　吉田　博之(22)福島県

おふたりとも，不幸な体験でしたね。秋こそはおいしいものを食べて幸せになってくださいね。

◆夏になると人気が出て困ってしまいます。とはいっても，相手が蚊では少しも嬉しくありません。すでに数え切れないほど，やられてしまいました。今年もまた蚊に嫌われる方法を考えている間に夏が終わることでしょう。

藤原　彰人(22)岡山県

どうやら編集部にもよくないムシがいるようです。赤いキスマークをつけられちゃって……。いえ，ホントの虫の話ですよ。

◆私はバイオリン弾きです。最初「Creative Computer Music入門」が始まったときは，あまりの簡単さに笑ってしまいました。でも今は，「う〜ん，そうなのか」と感動の連続です。自分の白痴ぶりを痛感している私は，またOh!XでDōGAを見てしまい，深みにはまっていくのでした。

前田　和彦(20)兵庫県

パソコンって，奥の深い可愛い奴ですよね。一緒に深みにはまりましょう。ずぶずぶ。

◆そうか，実は文系の人も多かったのか。

加藤　寿彦(18)神奈川県

みんなにやさしいOh!Xです，なんてね。

◆懺悔します。妹の友達の名は「荻窪　圭」ではなく「荻窪　佳」の誤りでした(母の指摘で判明)。この場を借りて，皆様にお詫び申し上げます。さらに訂正しますと，うちの妹の名前は，「百合江」ではなく「有理絵」です(確か，ハガキにはひらがなで「ゆりえ」と書いたはずなのに)。以上，ごめんなさいのコーナー番外編でした。ぐしぐし(Ⓒ岩瀬貴代美)。

松本　拓司(18)埼玉県

ほんとにひらがなだったら，編集部からもごめんなさい。ぐしぐし(Ⓒ岩瀬貴代美)。

◆この頃勝手に買わせていただいてますが，この本を買うには，何か「資格」が必要なのでしょうか？　それとも「儀式」でしょうか？

村上　学(17)埼玉県

まず，用意するものは，生贄の山羊の血と編集部への貢ぎ物と……というのは嘘ですが，600円(あ，ディスクが付いてるときは780円ね)は必要だと思います。

◆最近，祝一平氏の記事が載りませんが，なにか深いワケでもあるのでしょうか？

1)　満開製作所のほうが忙しい
2)　なにかよからぬソフトを制作中
3)　編集部と仲が悪い！

以上のうちから答えよ(ときどきは記事を載せては……)。　　　　　稲垣　友信(37)京都府

個人的に「よからぬ」あるいは「めでたい」ことで忙しい，とか？

◆僕としては対戦ポピュラスIIを，再び祝さんVS西川さんでやってほしいと思うのですが，どんなもんでしょ？　それがだめなら，ほかの人たちみんなでトーナメント戦をやるというのはどうでしょう？　　　　　三沢　弘之(21)神奈川県

さあ，本命は誰でしょう？

◆「ねえパパ，お願いがあるの」「なんだね，ステレオ(MIDI)でも買ってほしいのかい？」「ちがうの，あたし，お店(HD)が欲しいの」「し，しかし最近おまえの部屋(RAM)を広くしたばかりだし……」「そんな！　もう，あたしのこと愛してないのね。どうせ，あたし(X子)なんか2号さんよ。こうなったら，あたしたちのこと，あなたの奥さん(PC江)に全部バラしてやるっ！」「ま，まってくれ！　それだけはっ！」<つづく>　　　　　福岡　裕和(23)茨城県

我が家では，X美が本妻で，PC恵が2号だもんね。えっへん。

◆災いは急にやってくるものです。カーブで対向車が荷崩れして，あっ，と思ったときにはドカン！　車は大破しました。歩行者が歩いていたら死んでたでしょう。荷崩れとは……。大きな盲点でした。皆さんも気をつけましょう。

星野　大(20)新潟県

大変でしたね。怪我はしませんでしたか？

まさか，これは病院からのハガキでは……。それにしてもいったい何が崩れてきたのですか。

◆ASK68Kと付属の辞書で，全角で「X68000」とやって変換すると，「X68000」のほかに「何か御用ですか」「私がX68000です」と出ること，知ってましたか。　　　　　進戸　健太郎(16)兵庫県

おちゃめな辞書でしょ。知らなかった人は試してみてね。

◆8月号の話。某所4階の書籍コーナーでOh!Xを見ていると，ゼクセクスの楽譜とドラゴンセイバー(写真はSt.4だったでしょ)のプログラムリストが載っていたのでラッキーだなと思っていたら，右で「ゼクセクスにドラセイだって」という声がした。見ると，悪友2人がにーっと笑っていた。どうして類は友を呼んでしまうのでしょう(単に発売日だったからじゃないのか？)。でもパフェをごちそうになったので，やっぱりラッキー！　　　　　岩瀬　貴代美(20)福岡県

◆8月号の81ページについて。私のような初心者がいることも忘れないでくださいね。X68000って，ハードは親切だけれどソフトは無愛想。わからないことがあっても，近くで持っている人いないし。大学の生協に貴誌はたくさん置いてあるけれど，買ってる人は見たことありません。仲間が欲しい。　　　　　佐藤　夕子(21)神奈川県

類は友を呼ぶのです。生協とか町なかの本屋さんなどで「張り込み」をしていれば，きっと仲間は見つかるでしょう。

◆私は通学途中の本屋でOh!Xを買います。夏休みで学校に行かないと，入手方法がないよ〜。

冨田　英明(17)愛知県

そんなこと言わないでOh!Xを買いに行ってくださいね。

◆この前，学校からの帰り道，友達と話しながら歩いていたところ，どこからともなくおばさんが現れ，僕らの前に立ちふさがったかと思うと，両手の親指をこっちに突き出し，「ドン！」と言って，スタスタ走って行ってしまいました。あまりのことに，僕ら2人はなにごともなかったように，その前まで話していた話の続きを話し始め，ドンおばさんについては決して触れようとはしませんでした。　　　　　志田　健(16)東京都

一緒に遊んでほしかったとか……？

◆先日(7月28日)，海に連れて行かれました。夜中1時に電話がかかってきて，人のいい僕は二つ返事でOK。3台の車で6人で行ってきましたが，むこうに着いたのは，朝の4時。「まだ日が昇ってないぞ」とか「人が誰もいないぞ！」などといいつつ，しっかり遊んできました。これで全身まっ赤に日焼けして，その日の夜に入った風呂は拷問のようでした。

増田　雅光(20)愛知県

夏が満喫できてよかったですね。6人で3台ということは，カップル3組で行ったのかな？

◆ついに，うちのX68000もフロッピーをイジェクトしなくなった。しかし，ファイナルファイトのディスクだったので，分解して取り出した。

◀橋本　和典　東京都
うっ，クチビルがイロっぽい！　もう一枚の水着の女の子も可愛かったんだけど，この表情に負けて，こっちを載せちゃう担当者であります。

◀岡村　直也　兵庫県
「なかよし」姉弟再び登場。さて，おちゃめな姉ちゃんのプレッシャーに負けず，「天才プログラマ（3月号を見てね）」弟くんの合格なるか？

またディスクを入れると……出てこない。しかたがないので，シャープに電話した。ところが，イジェクトボタンを数十回押すと……イジェクトされた。どーしよ。シャープが家に来るまで，あと2日。また壊れるかな？(いまは正常に動いているんだよー)　　坂本　一樹(19)茨城県

風邪をひいても，注射されるのが怖くて，病院に行ったとたんに熱が下がっちゃったなんてことありませんでしたか？　パソコンも修理されるのを怖がっているのかもしれません。

◆「CD-ROM Drive for X68000売り上げ倍増計画」。DRIVE，メモリ，そして文月さんの「TORNADO」をはじめとするCGA作品を収めたCDをセットにして売るのだ！　もちろん，店頭でのデモもする。T◯WNSのフラクタクル……と違って，ユーザーの作品ってところも，ポイント高いぞ。販売が無理でも，デモするだけでX68000の底力が一般の人々にも示されるであろう。きっとメモリも爆発的に売れるぞっ。

たかはし　だいすけ(21)愛知県

デモを見た一般の人々の間で大評判になったX68000が爆発的に売れて，ついでにOh!Xも爆発的に売れる……といいなあ。

◆このたび，CATVに加入しました。衛星放送が見られるのがなんといってもよいですね。あとは「ウルトラQ」の再放送をしてくれればいうことなし。こんな私は，「なに，配属されてから2カ月しかたっていなかったのか」と，周囲に驚かれるほど職場に馴染んだ新入社員です。あはははは。　　宮岡　三幸(25)神奈川県

このハガキが載る頃には職場の「ヌシ」になっているのでは。

◀上田　考一　福岡県
濡れてるみたいな髪と，「へこたれないぞっ」て感じの表情がいいですね。X68000のパワーのイメージかな。

ぼくらの掲示板

●掲載ご希望の方は，官製ハガキに項目(売る・買う・氏名・年齢・連絡方法……)を明記してお申し込みください。
●ソフトの売買，交換については，いっさい掲載できません。
●取り引きについては当編集部では責任を負いかねます。
●応募者多数の場合，掲載できない場合もあります。
●紹介を希望されるサークルは必ず会誌の見本を送ってください。

仲間

★このたびX1/turboシリーズユーザーを対象とするサークル「Lovers」を発足しました。それにともない新規会員を募集します。活動は2カ月に1回のディスクマガジンの発行を予定しており，ミュージック，ゲームそのほかのプログラムの掲載をはじめ，CGや会員のフリートークなどもあります。X1で何かやりたい，みんなとコミュニケーションをしたい，と思っている人は，ぜひ私たちのサークルにご連絡ください。なお，入会希望の方は案内書を送付いたしますので，62円切手を同封のうえ下記の住所まで送ってください。〒646　和歌山県田辺市駅前新通り959　中川　学

★「OREGA」では新規会員を募集します。当クラブでは年6回の会報発行を中心に活動しています。会報にはプログラミング講座，ハードウェア講座，ゲーム，パソコン通信，体験談，SF，イラスト，マンガなど，コンピュータの周辺にある面白いことを幅広く載せています。入会希望の方は案内書をお送りしますので，124円分（62円×2）の切手と郵便番号，氏名，住所を記入したラベル，または封筒を同封して下記の住所までお送りください。〒910　福井県福井市文京4－9－5　メゾン山本201　新海敏之方「OREGA」案内X係

売ります

★X68000用数値演算プロセッサを21,000円で売ります。連絡は往復ハガキでお願いします。〒062　北海道札幌市豊平区平岸一条12丁目　コーポ目黒314号室　瀬川　正博(28)

★Roland「CP-40-7」（はなうたくん3.5インチ版）を送料込み17,000円で売ります。箱，取説，保証書すべてあり。新品同様。連絡は往復ハガキでお願いします。〒253　神奈川県茅ケ崎市出口町9-3　原　明(43)

★シャープ漢字ドットプリンタ「CZ-8PK5」を30,000円で売ります。プリンタケーブルなし，マニュアル，箱あり。連絡は往復ハガキで。〒940　新潟県長岡市東神田3-4-12　伊藤　宏幸

★X1/X68000用136桁ドットインパクトプリンタ「CZ-8PK6」を20,000～35,000円で売ります。箱，マニュアル，付属品あり。用紙も付けます。また，X1用拡張I/Oボックス「CZ-8IEB」を希望価格で売ります。箱，マニュアル付きです。それぞれ，希望価格を書いて往復ハガキで連絡してください。〒510-12　三重県三重郡菰野町菰野2938-3　神田　武幸(19)

★熱転写カラー漢字プリンタ「CZ-8PC5」を40,000円で売ります。マニュアル，付属品はありますが，箱はありません。数回使用の美品です。連絡は往復ハガキでお願いします。〒245　神奈川県横浜市泉区和泉町6228-21　後藤　智博(20)

★熱転写カラー漢字プリンタ「CZ-8PC5」を45,000円で売ります。また，RolandのMIDI音源モジュール「CM-64」＋カード2枚を65,000円で売ります。連絡は電話番号を明記のうえ，官製ハガキでお願いします。〒381-02　長野県上高井郡小布施町851　鈴木　理星(17)

買います

★MZ-2000/1500/700用FDDインタフェイス「MZ-1F07」を2,000円，X1用データレコーダ「CZ-8RL1」を5,000円で買います。連絡は往復ハガキでお願いします。〒636-03　奈良県磯城郡川西町唐院213　吉仲　正和

★RolandのMIDI音源モジュール「MT-32」を付属品，マニュアル付き，30,000円で買います。連絡は官製ハガキでお願いします。〒739-01　広島県東広島市八本松町飯田383-2　広明寮　横田　紀明(25)

★RS-232Cインタフェイス「MZ-1E24」を5,000円程度で買います。連絡は往復ハガキでお願いします。〒050　北海道室蘭市水元町36-12　小川ハイツ205　小川　幸泰(19)

★X1シリーズ用5インチ2D/2HD兼用ディスクドライブ「CZ-520F」を25,000円程度。または，5インチ2Dディスクドライブ「CZ-502F」を20,000円程度。5インチ2Dディスクドライブ「CZ-503F」なら15,000円程度で買います。連絡は官製ハガキか封書に売ってくれる機種を明記して，お願いします。〒302-01　茨城県北相馬郡守谷町守谷甲2779-109　高橋　顕治(25)

バックナンバー

★Oh!MZ1986年8月号，1987年1月号，Oh!X1990年6月号を各1,500円（送料込み）で買います。切り抜きのあるものは不可。連絡は官製ハガキでお願いします。〒546　大阪府大阪市東住吉区田辺4-12-10　鶴身　哲郎(20)

★Oh!MZ1987年2月号，4月号を送料込み各1,000円で買います。切り抜き不可。連絡は官製ハガキか封書でお願いします。〒302-01　茨城県北相馬郡守谷町守谷甲2779-109　高橋　顕治(25)

編集室から from E・D・I・T・O・R

DRIVE ON

このコーナーでは，本誌年間モニタの方々の意見を紹介しています。今月は8月号の内容に関するレポートです。

●8月号でいちばん印象に残ったものは「大人のためのX68000」でした。私は，以前某BBSでの詐欺事件が世間を騒がせてから，パソコン通信をしていません。私の入っているBBSではありませんが，他人事とは思えなかったのです。そして，熱が冷めてしまいました。楽しいのも便利なのもわかります。しかし，なぜ他人の楽しみを壊すようなことをするのでしょう。いろいろな可能性のあるメディアなのに悲しいことです。でも，この記事を読んでまたチャットをしたくなりました。そうなると電話代というのも大きな問題ですが……主人に残業してもらおう。

野原　志貴乃(30)　X68000 ACE-HD　埼玉県

●ゲームでもそのプログラムに対し，テクニカルな疑問と欲求をもつ。そして，その疑問や欲求を解決，あるいは満たそうとする野望が生み出される。その野望がいかに大きくても必ず，どこかに達成する策が存在するのだ。僕は，横内さんの「ちょっと大きなモノを書こう」を読んでそう思いました。金も根性もない自分に，まだ足りないものがあったのかと思うと悲しくなります。しかし，それらを満たせばどうにかなる，と思うとまだ道は開放されているのでしょうね。希望の光はまだ消されていないのです。

林　寛(18)　X68000 EXPERT　三重県

●8月号の特集を初めに読んだときには，なにかまとまりがない印象を受けました。しかし，よく読み進めていくと中森氏の「プログラミング言語の前に」を受けて記事が展開しているような感じでしたね。中森氏の記事では，データ構造についてもう少し具体的な例を挙げて説明してもらいたかったところですが，特集のタイトルがなぜ「プログラミング再入門」なのかわかったような気がしました。あと，新製品紹介にあった「MATIER」は，これまでで最も「使えそうだ」という思いを抱かせるグラフィックツールだと思いました。「Z's STAFF PRO-68K」がどんなに高性能でも，このソフトを使って私になにが描けるか？　というと非常に疑問なのです。しかし，「MATIER」の記事を読んでいると不思議にも手軽になにか描けるのではないか，という気がしてきたのです。決して，気のせいであってほしくありません。もちろん絵を描くなら絵心が必要ですが，せっかくパソコンを使っているのだから肉体的な作業とは違ったアプローチで絵を描けるツールであったらいいな，そう思っています。

矢野　啓介(19)　X68000 XVI,MZ-2500 北海道

●8月号で気になった記事は「LIFE110.X」でした。いままでは，ライフゲームなんて適当に点を打って眺めていると，すぐに動きが止まっちゃうだけじゃん。そう思っていたのです。しかし，この記事を読んでみるとライフゲームにもすごいものがある，と感じてしまいました。「LIFE110.X」に少しウェイトを入れて眺めていると，なんだかディスプレイの中に生き物がいるように思えてきます。やはり人間というものは，自分の知らないことを知ったときには喜びを感じるものなのですね。基本的なことから蘊蓄(うんちく)めいたことまで，普通の人生では知ることのできないようなことを教えられる「LIFE110.X」が好きになりました。また，この記事を読んだだけでライフゲームがすべてわかったような気にさせてくれる，そんな調子がよかったです。

加藤　恵吾(16)　X68000 SUPER　愛知県

●私はあまりプログラミングをしません。その理由は，とっつきにくさ，わかりにくさがあるからだと思います。とっつきにくさは面倒臭さでこれが大きな壁になっています。わかりにくさは，複雑だということでしょう。時間がないという考えはもっていません。本当に楽しければ，時間を作ることは簡単だと思うからです。しかし，プログラムを組むことはパソコン本来の利用法だと思っているので，明日とはいわず今日からでも始めてみたいですね。また，8月号の特集の記事の中では，文月さんの「プログラムの流れをつかもう」がよかったです。なんともいえなく面白い，のひと言です。文章もカットも補足も(このへんがほかのライター方と決定的に違うと思う)です。おそらく文月さんは，CGAのプログラムも文章もセンスよくまとめてしまうのでしょう。読者が読んで楽しめる文章であって，その内容の難しさが興味に変わっていくのがはっきりわかります。

村上　晃(23)　X68000 XVI　岡山県

●8月号の特集「プログラミング再入門」は，具体的なプログラムの組み方や作法よりも，心得や思想といった概念的な記事が多かったですね。サンプルプログラムは満足のいくものですが，現在ではプログラミング言語もいろいろと出回っています。それらの言語のマニュアル的なものがあればもっとよかったです。また，新製品紹介の「TG100」がよかったですね。ヤマハも本格的にDTMへ力を入れてきた証でしょうか。まず，45,000円という価格だけで，「買い」と思ってしまいます。GMモードやCM-64モードを備えているのも，ユーザーにとってはかなり心強いことでしょう。

藤田　康一(21)X68000 PRO　静岡県

●特集では「正しい花瓶の落とし方」がよかったです。これを読んで，BASICもうまく使えば十分役立つということがわかりました。物理現象を理解する力とBASIC，そしてちょっとした考え方（ここが難しい）をすれば，実現可能ということを教えられました。「SOUND SX-68K」は，従来の機能をそのままもってきたものであまり進歩が見られない，そう感じました。なにか光るものを見つけてほしいです。そうしないとほかのソフトにつぶされてしまうでしょう。

酒本　一幸(19)　X68000　千葉県

ごめんなさいのコーナー

8月号　P.56　ゲーム内部のイロハ

リスト1の6, 16行に誤りがありました。正しくは以下のとおりです。

```
 6：  move.w  (a1),d0
16：  lea.l  data_size(a1),a1
```

9月号　P.74　夏休みの最小2乗法

リスト1に誤りがありました。以下のとおり修正してください。

```
410 ……,512-y(i)*y1/dy-y0……
```

バグに関するお問い合わせは

☎03(5488)1311(直通)

月〜金曜日16：00〜18：00

お問い合わせは原則として，本誌のバグ情報のみに限らせていただきます。入力法，操作法などはマニュアルをよくお読みください。

また，よくアドベンチャーゲームの解答を求めるお電話をいただきますが，本誌ではいっさいお答えできません。ご了承ください。

なにをしたい なにができる なにを必要とする

▶今回の特集では，X68000のDTM環境，特にZ-MUSICを中心にしたシステムを紹介しました。Z-MUSICを音源ドライバ単体として使用するだけでなく，Z-MUSICを中核にして統合した音楽環境を構築していこう，そんな思いを感じることができたと思います。

　また，わかりきったことですが，別にMIDI音源がなければDTMができないわけではありません。X68000ならば内蔵のFM8音＋ADPCMでかなりの表情を持った音楽が作れます。必要なのは素材を生かす感性。現在では，個人レベルでもプロと同じ機材を手に入れられる時代です。自分にあった楽器を手に入れ，自分にあったシステムを構築することが大切でしょう。束縛なしに思ったとおりの音楽を作れる，そんな環境を目指したいものですね。

▶やるといいながら，ずいぶんと時間がたってしまったOh!X Booksシリーズ。Z-MUSICのほかに，SX-WINDOW，グラフィック，そしてCARDDRV.Xを使ったカードゲームシリーズなど。計画中のものはこのくらいですが，どのようなものを配布するかはまだまだ流動的。現在でも投稿は受けつけていますので，なにかやってやろう，と思った人はぜひ投稿しましょう。特にオリジナルカードデータ，カードデータ作成支援ツールは狙い目です。気力のある人は挑戦してみてください。

▶そして，遅れてしまいましたが待望のX68000用CD-ROMドライブ使用レポート。最近流行のマルチメディアという言葉に刺激されてか，アンケートハガキの中にもかなりの期待をしている人が見られました。いままでも，個人レベルで実験的にCD-ROMは使われてきました。しかし，メーカーからサポートされたことにより，安心してCD-ROMを使える環境への道が開けてきたのではないでしょうか。この大容量メディアが，X68000の世界の中でどのような位置付けを得るのか。興味深く見守っていきたいものです。

▶猫とコンピュータ，知能機械概論はともに著者多忙のためお休みさせていただきます。

投稿応募要項

●原稿には，住所・氏名・年齢・職業・連絡先電話番号・機種・使用言語・必要な周辺機器・マイコン歴を明記してください。

●プログラムを投稿される方は，詳しい内容の説明，利用法，できればフローチャート，変数表，メモリマップ（マシン語の場合）に，参考文献を明記し，プログラムをセーブしたテープ（ディスケット）を添えてお送りください。また，掲載にあたっては，編集上の都合により加筆修正させていただくことがありますのでご了承ください。

●ハードの製作などを投稿される方は，詳しい内容の説明のほかに回路図，部品表，できれば実体配線図も添えてください。編集室で検討のうえ，製作したハードが必要な場合はご連絡いたします。

●投稿者のモラルとして，他誌との二重投稿，他機種用プログラムを単に移植したものは固くお断りいたします。

あて先

〒108 東京都港区高輪2-19-13 NS高輪ビル

ソフトバンク出版部

Oh!X「㋢㋮㊔」係

S H I F T · B R E A K

▶週末になると秋葉原で出物を発掘する探索作業が習慣になっている。なんといっても，ツボにはまると珍品が破格の値段で入手できるのが魅力だ。唯一のルールは，迷わずに衝動買いすることなのだが，これが身についたおかげでふだんでも衝動買いが増えて万年金欠。もっと仕事しなくっちゃね，うんうん。（八）

▶E.O.さんの送別会（?）が編集部近くの飲み屋であった。ああ，思えばあっという間の3年間だった。長い間ご苦労さまでした（感謝）。早くミュージシャンとしてデビューしてCDを送ってくださいね。しかし久しぶりに飲んで熱く語ってしまった。もうそんなエネルギーは残っていないと思ってたからちょっと嬉しかったかな。うん。（哲）

▶なにげなく軽井沢に行こうと思い立ち，3日後には軽井沢にいた。出発前には江戸川土手でバーベキューをやり，午前3時半には軽井沢駅前にいて，午前6時にはチーズケーキ屋の前で仮眠をとっていた。渋滞にハマル前に帰ったのだが，屋根を開けて走っていたので焼けてしまったのはつい2日前。また明日も軽井沢に行く私っていったい……。（S.K.）

▶現在，MIYA-NETでZ-MUSICのサポートをしてくれています。最新版のZ-MUSICやZ-MUSIC支援ツール，演奏データなどがアップされています（ゲストもダウン／アップ可）。電話番号048（648）9801（Tri-P:CXMIYA），入会無料，24時間（ゲストはPM9：00～AM1：00は利用不可）。あと，MIYA-NETのテレカを5名にプレゼントします（プレゼント番号0）。（善）

▶女友達に素敵な石を贈った。というと誤解を招きそうだが，女友達を某国語に訳して石の名前が数字とくれば，いつもどおりの自慢話。一部のソフトは強烈に速くなり，一部のソフトは暴走する。高速化の改造は，ハード・ソフトともに大きなリスクを覚悟したうえで使わなきゃ。動かなければあきらめる。保証など期待しているうちは甘いぜ。（A.T.）

▶NHKが私の生活パターンに合わせて放映してくれたおかげで，今年はオリンピックとやらに馴染んでしまった。掘り出しものは，なんといっても吉田の内股である。奥襟をつかむ。足が小刻みに動く。誰でもわかる内股狙い。何度か躊躇し，踏み込んで蹴り上げ高く持ち上げる。見え見えの内股をかけて完璧に投げる。最近貴重な豪快さだ。（K）

▶ああ，ついに。仕事が納期遅れになりそうだったので某98ノートを買った。もちろん休日に仕事をするためだ。僕も並の人間になったかと思うとちょっと悲しい。なんとか仕事を終えたいまでは，ワープロ1本しか入っていない120Mバイトのハードディスクが妙にさみしい。このままほこりをかぶってしまうのは目に見えてるし，早まったかな。（KO）

▶8月に隣の編集部から引っ越してきた。いちばん喜んだ（？）のは，あたしの「さふぁいあ」ちゃん（あ，X68000ね）だろう。「お絵描きしたい」とか「プログラミングしたい」とかいって買ったのが1年くらい前。なのに，ゲームばっかでちっとも遊んであげてなかったもんねぇ。さっそく会社に連れてきて環境設定とか勉強中。ああ，しあわせ。（ふ）

▶夜を徹して飲んだくれたり，海へ行ったり，夜中に車を飛ばしてボウリングをしたり……。なんだかんだいって，今年の夏も結構充実していたなあ。てなもんで，ああっそんなところに沼を作るんじゃない。ひでえ，リーダーを狙うなんて卑怯である。返り討ちだ！　返り討ち。くっそう，右肩がこるじゃねえか。（J）

▶今年も誕……，まあいいや。CD-ROMのテストを眺めていて思ったんだけど，CDTVにSCSIがついていればもっと売れただろうなあ。MacintoshやX68000（要ドライバ）につなぐのだ。160,000円ぐらいと少し割高になるけれど，キーボードとフロッピードライブをつなぐとAMIGA500になると思えば安いもの。なぜかMIDI端子もついているし。（A）

▶ったく，Macユーザーって○○で△△だよなあ。話変わって，私の読書歴で最悪だった本はヒルティの『眠られぬ夜のために（上）』というヤツだ。聖書のことごとくを正当化し，返す刀で一方的な見地で他人の思想を誹謗する。で，論拠はほとんど聖書という……。自己の正しさに一点の疑いもないことが，無神論者には醜悪に映るのだろうか。（U）

▶E.O.嬢を送り出し，Oh!Xはまたひとつの時代を終えた。E.O.さん元気でね。さて，入れ替わるようにOh!X入りしたのがF.T.嬢。お隣のOh!PCからの異動だが，これがまたX68000XVIユーザーなのだ。しかも自分のマシンをサファイアちゃんなんて呼んでいる。けっこうきてるな……。話は違うけど，上のU氏の話がわかる人ってすごいと思う。（T）

microOdyssey

丸子橋までは10分で出られる。そこからは快適な走行が続く。車もいない。信号もない。歩行者に対してまでも，“サイクリング走行にご協力ください”という標識が立てられている。どう見てもはっきりと歩行者用道路とわかる道なのに，たまにバイクで走っている大馬鹿ヤローがいるが，そういうやつが対向してきてももちろん避けてなんかやらない。土手から落ちてしまえばいいのだ。

多摩川サイクリングロードがどういう経緯で作られたのかは知らない。アスファルトで土手の上を固めてそんなものを作る前に，川をきれいにしたり，やることはいっぱいあるだろう。それに，歩行者よりも自転車が優先されるというルールにも少し首を傾げる。

しかし，そこを自転車で走っていると，ただ気持ちいい。

アップダウンは少ないし，一般道路が並行しているわりに排気ガスも漂っていない。

普通の自転車（ママチャリってやつ）でのんびり走っているおじさん，おばさん。三輪車を一生懸命こいでいる子供。じゃれあいながら散歩している親子やペット連れ。

ロードレーサーで真剣に走行している人ももちろん多い。サラリーマンの日曜サイクリストとかはいいとして，レース走行の練習みたいに猛スピードで飛ばしていくおにいちゃんたちにはちょっと目を釣り上げてしまう。ちゃんと周りの歩行者に気を使っているのだろうか。気配りしているなら別にいいのだが。

こちらは体力もないし，足の筋肉を太くしたくもないので，いつも時速25kmぐらいで流している。下り坂もないので，飛ばしても時速40km前後。

通算走行距離は2カ月がたってやっと500kmに達した。このうち多摩川以外で走ったのは，鎌倉への片道走行ぐらい。1週間にほぼ50kmのペースになるが，休めない土日がたまにあることや，はじめの1カ月が梅雨であったことを考えると，自分では納得のいく距離だ。

さて，時速や走行距離がわかるというのは，スピードメーターをつけているからだと察せられるだろう。そのとおりなのだが，それはサイクルコンピュータと呼ばれる電子機器である。

センサーは非接触式で，前輪のスポークにつけられたマグネットを探知し，回転速度を弾き出す。で，あらかじめ入力してあるタイヤの半径から現在の速度を算出するというわけだ。そこから平均速度や走行距離も計算できる。

タイマーから時間，センサーから回転速度を拾ってきて，あらかじめ入力された数値を元に速度を計算しているのは，4ビットマイコンチップ。サイクルコンピュータという名称は少しおおげさな気もする。サイクルコンピュータというからには，自分の体調や天候も考慮のうえ，“××Kmで走りなさい”とでもいってくれなくっちゃあ，という向きもあるだろう。

しかし，それでは興醒めである。速度と距離だけが表示される小さなパネルを見ながら，自分でペースを作っていけばいいのである。

コンピュータはいろいろなところに組み込まれて，なんでも便利にしてくれている。が，程度をわきまえた使い方というのも必要ではないだろうか。ここは4ビットで十分なのだ。 (A)

1992年11月号10月17日(土)発売

特集　ゲームマネージメント
・ゲームにおけるキャラクタ管理
・専用マネージャの考え方

速報　データショウ'92

対戦ポピュラスIIのすべてを探る

X68000用画面保存ドライバ　SAVESC.SYS

全機種共通システム
簡易テキストエディタ

バックナンバー常備店

東京	神保町	三省堂神田本店5F	03(3233)3312
	〃	書泉ブックマートB1	03(3294)0011
	〃	書泉グランデ5F	03(3295)0011
	秋葉原	T-ZONE 7Fブックゾーン	03(3257)2660
	八重洲	八重洲ブックセンター3F	03(3281)1811
	新宿	紀伊国屋書店本店	03(3354)0131
	高田馬場	未来堂書店	03(3209)0656
	渋谷	大盛堂書店	03(3463)0511
神奈川	横浜	有隣堂横浜駅西口店	045(311)6265
	〃	有隣堂ルミネ店	045(453)0811
	藤沢	有隣堂藤沢店	0466(26)1411
	厚木	有隣堂厚木店	0462(23)4111
	平塚	文教堂四の宮店	0463(54)2880
千葉	柏	新星堂カルチェ5	0471(64)8551
	船橋	リブロ船橋店	0474(25)0111
	〃	芳林堂書店津田沼店	0474(78)3737
	千葉	多田屋千葉セントラルプラザ店	0472(24)1333
埼玉	川越	黒田書店	0492(25)3138
	川口	岩渕書店	0482(52)2190
茨城	水戸	川又書店駅前店	0292(31)0102
大阪	北区	旭屋書店本店	06(313)1191
	都島区	駸々堂京橋店	06(353)2413
京都	中京区	オーム社書店	075(221)0280
愛知	名古屋	三省堂名古屋店	052(562)0077
	〃	パソコンΣ上前津店	052(251)8334
	刈谷	三洋堂書店刈谷店	0566(24)1134
長野	飯田	平安堂飯田店	0265(24)4545
北海道	室蘭	室蘭工業大学生協	0143(44)6060

定期購読のお知らせ

Oh!Xの定期購読をご希望の方は綴じ込みの振替用紙の「申込書」欄にある『新規』『継続』のいずれかに○をつけ，必要事項を明記のうえ，郵便局で購読料をお振り込みください。その際渡される半券は領収書になっていますので，大切に保管してください。なお，すでに定期購読をご利用の方には期限終了の少し前にご通知いたします。継続希望の方は，上記と同じ要領でお申し込みください。

海外送付ご希望の方へ

本誌の海外発送代理店，日本IPS(株)にお申し込みください。なお，購読料金は郵送方法，地域によって異なりますので，下記宛必ずお問い合わせください。

日本IPS株式会社
〒101 東京都千代田区飯田橋3-11-6
☎03(3238)0700

Oh!X 10月号
■1992年10月1日発行　定価600円(本体583円)
■発行人　孫　正義
■編集人　橋本五郎
■発売元　ソフトバンク株式会社
■出版事業部　〒108 東京都港区高輪2-19-13　NS高輪ビル
Oh!X編集部　☎03(5488)1309
出版営業部　☎03(5488)1360 FAX 03(5488)1364
広告営業部　☎03(5488)1365
■印　刷　凸版印刷株式会社

©1992 SOFTBANK CORP. 雑誌 02179-10 本誌からの無断転載を禁じます。
落丁・乱丁の場合はお取り替えいたします。

今冬発売！

ユーザーのみなさん、お待たせしました。

X68000上でのプログラミング環境を充実させるために、
各種開発ツールキットをテーマ別に解説。
それが

〈 X68k Programming Series 〉
X68k プログラミング シリーズ

です。

X68k Programming Series ♯1 Develop.

吉野智興・中村祐一・石丸敏弘・今野幸義　共著
Ｂ5判・箱入り　2冊セット/ディスク付　予価9,000円(税込)　11月上旬発売予定

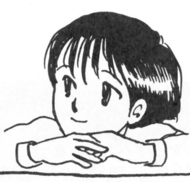
本書は、X68000用に移植されているCコンパイラX68000 GCC(GCC)、アセンブラHigh speed assembler(HAS)、リンカHigh speed linker(HLK)、デバッガGNU Debugger(GDB)について新たに書き下ろしたドキュメントであり、開発キットです。付属ディスクにはこれら4種類の開発キットとサンプルプログラムが収録されています。またライブラリには、XCコンパイラおよび同シリーズの『libc』のライブラリが利用できます。
「Vol.1　Programmer's Guide」「Vol.2　Reference」の2冊より構成。Vol.1では、基礎知識やインストール方法、そしてGCC、HAS、HLK、GDBの各機能および操作方法について解説します。そしてVol.2は各種オプションスイッチやエラーの対処方法についてまとめられており、ハンディマニュアルとして最適です。

X68k Programming Series ♯2 libc

村上敬一郎・荻野祐二・大西恵司　共著
Ｂ5判・箱入り　2冊セット/ディスク付　予価8,500円(税込)　12月中旬発売予定

本書は、Cコンパイラで利用可能なライブラリ関数を5つのレベルに分類、それぞれよく使用される約650種類のライブラリ関数を著者らが独自に開発し、その解説を書き下したものです。作成したライブラリ関数はすべて付録ディスクに収録されています。これらのライブラリ関数はＸ68000上で動作し、XCコンパイラおよび同シリーズの『Develop.』で使用することができます。
「Vol.1　User's Reference」「Vol.2　Programmer's Reference」の２冊より構成。Vol.1ではライブラリ関数を使用していく上での基礎知識や注意事項、およびファイル操作やユーザー管理などのライブラリ設計について解説していきます。Vol.2は付録ディスクに収録されている関数について、サンプルプログラム入りで解説したマニュアルとなっています。

SOFT BANK ソフトバンク出版事業部



満開の電子ちゃん

作・え　岡村祭

講読方法：定期購読もしくはソフトベンダーTAKERUでお買い求めいただけます。

★定期購読の場合＝購読料６ヶ月分6,000円（送料サービス、消費税込）を、現金書留または郵便振替で下記の宛先へお送り下さい。

現金書留の場合：〒171　東京都豊島区長崎1-28-23 Muse西池袋2F　㈱満開製作所

郵便振替の場合：東京　５－362847　㈱満開製作所

- ご注文の際は、郵便番号・住所・氏名・電話番号を忘れずに記入して下さい。
- 3.5インチディスク版をご希望の方は、「3.5インチ版」とご指定下さい。
- 新規購読の方は「新規」と明記して下さい。なお、特に購読開始号のご指定がない場合は既刊の最新号からお送りいたします。
- 製品の性格上返品には応じられませんが、お申し出があれば定期購読を解約し残金をお返しします。

★TAKERUでお求めの場合＝１部につき1,200円（消費税込）です。

- 定期購読版と内容が一部異なる場合があります。御了承下さい。
- お問い合わせ先　TEL(03)3554－9282（月～金　午前11時～午後６時）

（なお、定期購読版のバックナンバーについては定期購読の方のみご注文を承ります）

わたしは元は折り目正しい98ユーザーでした。それが友人の前で「クロックが上がったらX68000を買う」と口走ったばかりに、音と色のない生活には戻れなくなり、おまけに、その友人に勧められ、わけもわからないまま、電脳倶楽部というディスクマガジンの定期購読まではじめてしまいました。

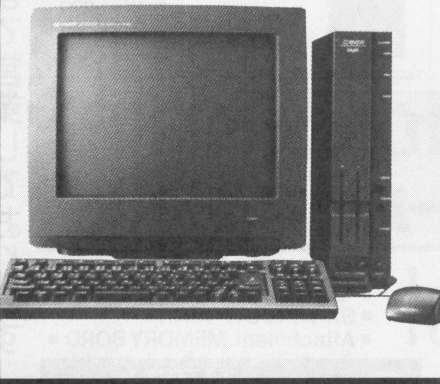
ところが、送られてきたディスクは電源オンですぐ起動、便利なツールが満載で、すっかり虜となり今では正しいX68000ユーザーとなれるよう精進する毎日を送っています。

八木　猛（神奈川県）

マイコンショップ川口

☎0482-25-1718

（消費税別）



New X68000 COMPACT XVI

~~¥298,000~~

CZ-674C-H	¥298,000
CZ-608D-H	¥ 94,800
AV-090-SC	¥168,000

定価 ¥560,000
超特価

ソフト各種超特価ご奉仕中

CZ-219SS OS-9/X68000	定価¥29,800
CZ-213MS MUSIC PRO68K	定価¥18,800
CZ-214MS SOUND PRO68K	定価¥15,800
CZ-215MS Sampling PRO68K	定価¥17,800
CZ-220BS DATA PRO68K	定価¥58,000
CZ-224LS The 福袋 Ver2.0	定価¥ 9,980
CZ-225BS Multiword	定価¥32,000
CZ-251BS Hyper word	定価¥39,800

中古売買価格表

品名	買取り価格	売価
CZ-633C	160,000より	180,000より
CZ-644C	210,000より	230,000より
CZ-613C	105,000より	125,000より
CZ-603C	75,000より	95,000より
CZ-612C	85,000より	98,000より
CZ-602C	65,000より	85,000より
CZ-653C	75,000より	95,000より
CZ-663C	95,000より	115,000より
CZ-662C	75,000より	98,000より
CZ-652C	55,000より	75,000より
CZ-611C	70,000より	89,000より
CZ-601C	45,000より	65,000より
CZ-612D	35,000より	45,000より
CZ-602D	30,000より	39,800より
CZ-603D	20,000より	29,800より
CZ-604D	25,000より	34,800より
CZ-605D	45,000より	55,000より

プリンター

CZ-6VT1	特価¥ 47,700
CZ-8PG1	特価¥ 86,800
CZ-8PG2	特価¥106,900
CZ-8PK10	特価¥ 66,800
CZ-8NS1	特価¥141,000
CZ-6BC1	特価¥
CZ-6BG1	特価¥
CZ-6BP1	特価¥
CZ-6BP2	特価¥ 34,400

ラムボード

CZ-6BE2A	定価¥59,800	特価¥ 44,900
CZ-6BE2B	定価¥54,800	特価¥ 41,100
CZ-6BE2D	定価¥54,800	特価¥ 41,100
CZ-6BE1B	定価¥28,000	特価¥ 21,000
CZ-6BE2	定価¥79,800	特価¥
CZ-6BE4C	定価¥98,000	特価¥
PIO-6BE1-A	定価¥25,000	特価¥
PIO-6BE2-2M	定価¥50,000	特価¥
PIO-6BE4-4M	定価¥88,000	特価¥
SH-6BE1-1M	定価¥25,000	特価¥

ファイル

CZ-6MO1	定価¥450,000	特価¥
CZ-64H	定価¥120,000	特価¥
CZ-68H	定価¥160,000	特価¥

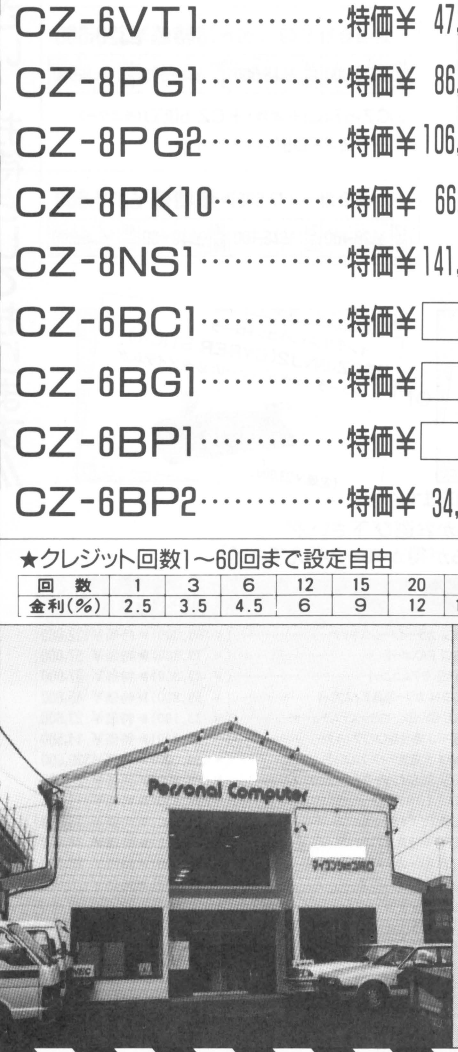
その他機種

CZ-8NS1	カラーイメージスキャナ	定価¥188,000	特価¥
JX-220X	カラーイメージスキャナ	定価¥168,000	特価¥
CZ-6BN1	スキャナ用パラレルボード	定価¥ 29,800	特価¥
CZ-6VT1	カラーイメージユニット	定価¥ 69,800	特価¥
CZ-6BV1	ビデオボード	定価¥ 21,000	特価¥
CZ-8TM2	モデムユニット	定価¥ 49,800	特価¥
CZ-8NJ2	インテリジェントコントローラ	定価¥ 23,800	特価¥
CZ-8NM3	マウス・トラックボール	定価¥ 9,800	特価¥
CZ-8NT1	トラックボール	定価¥ 6,888	特価¥
CZ-8NJ1	ジョイカード	定価¥ 1,700	特価¥
CZ-6BC1	FAXボード	定価¥ 79,800	特価¥
CZ-6BM1A	MIDIボード	定価¥ 26,800	特価¥
CZ-6BP1	数値演算プロセッサ	定価¥ 79,800	特価¥
CZ-6BP2	数値演算プロセッサ	定価¥ 45,800	特価¥
CZ-6TU-BK-GY	RGBシステムチューナー	定価¥ 33,100	特価¥

★クレジット回数1～60回まで設定自由

回数	1	3	6	12	15	20	24	36	42	48	54	60
金利（%）	2.5	3.5	4.5	6	9	12	12.5	17.5	22	23	28.5	29.5

中古品も取扱っております。



矢印の道路は一方通行です。

通信販売をご利用の方

―全国通販―

通信販売をご利用の方は、売値の変動がありますので在庫、値段をあらかじめ確認のうえ電話で、商品名及びお客様の住所・氏名・電話番号をお知らせ下さい。

■店頭にて、新作ゲームソフト25〜30%OFF!!（税別）、超低金利オクトハッピークレジットをご利用下さい!!

冬のボーナス一括〈10・11・12月末〉払いOK!!手数料無料!!ご利用下さい。

パソコンプラザ OCT オクト

オクトで始まるパソコンワールド

☎ 03-3730-6271

●営業時間 AM 11:00〜9:00/日曜・祭日PM7:00　電話一本で、ハイ即納

〒144 東京都大田区蒲田4-6-7 FAX 03-3730-6273

全国通販

定休日:毎週火曜日
（祭日の場合翌日になります。）

案内図

店頭セール実施中

OCT-1 システム インフォメーション

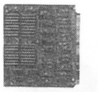
- ▶全商品保証付（メーカー保証）
- ▶超低金利ハッピークレジット（1回〜60回）頭金ナシOK!
- ▶ボーナス一括払いOK! ボーナス 2回・4回・6回 払いOK!
- ▶配達日の指定OK!（万全なサポート体制）
- ▶商品の組合せ自由! オクトフリーダムシステム
- ▶店頭デモンストレーション実施中

特選周辺機器（送料¥500）

■SX-68MII MIDIインターフェイスボード
（システムサコム）¥19,800… 特価**¥13,500**

■モデム〔FMMD-311G〕・2400bps.クラス5
（Fujitsu）¥35,800……… 特価**¥24,000**

■増設RAMボード＝I・Oデータ



① PIO-6BE1-A（1MB）¥25,000……… 特価**¥15,800**
② PIO-6BE2-2M（2MB）¥50,000……… 特価**¥31,000**
③ PIO-6BE4-4M（4MB）¥88,000……… 特価**¥54,000**
④ SH-6BE1-1M（1MB）¥25,000……… 特価**¥17,800**

OCT-1 蒲田

皆様方へ、残暑厳しい折の為!!
残暑お見舞いセール! 実施中!!

SHARP

X68000 Compact
PERSONAL WORKSTATION・XVI

■16MHz■
■SX-WINDOW ver1.1■
■Attachment MEMORY BORD■

■CZ-674C-TN（定価¥298,000）

Ⓐ●CZ-674C-TN
●CZ-608D-TN（14型カラーディスプレイ）

定価合計¥392,800▶ 超特価~~¥283,000~~

12回	24回	36回	48回
~~¥25,900~~	~~¥13,700~~	~~¥9,500~~	~~¥7,500~~

Ⓑ●CZ-674C-TN
●CZ-607D-TN（14型カラーディスプレイTV）

定価合計¥397,800▶ 超特価~~¥285,000~~

12回	24回	36回	48回
~~¥26,100~~	~~¥13,600~~	~~¥9,600~~	~~¥7,500~~

Ⓒ●CZ-674C-TN
●CZ-614D-TN（15型カラーディスプレイTV）

定価合計¥433,000▶ 超特価~~¥310,000~~

12回	24回	36回	48回
~~¥28,300~~	~~¥15,000~~	~~¥10,400~~	~~¥8,200~~

Ⓓ●CZ-674C-TN
●CZ-606D-TN（14型カラーディスプレイ）

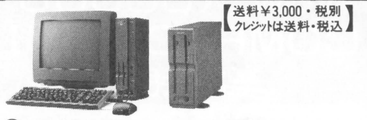
定価合計¥377,800▶ 超特価~~¥270,000~~

12回	24回	36回	48回
~~¥24,700~~	~~¥13,100~~	~~¥9,100~~	~~¥7,100~~

※送料¥2,000・税別
（クレジット価格は、送料・税込）

Compact限定セット

【送料¥3,000・税別 クレジットは送料・税込】



①CZ-674CH〈本体〉＋CZ-608DH〈モニター〉＋CZ-6FD5（5インチドライブ）

定価合計¥492,600▶ 超特価~~¥320,000~~

12回	24回	36回	48回
~~¥29,300~~	~~¥15,500~~	~~¥10,800~~	~~¥8,500~~

②CZ-674CH〈本体〉＋CZ-606D〈モニター〉＋CZ-6FD5（5インチドライブ）

定価合計¥477,600▶ 超特価~~¥310,000~~

12回	24回	36回	48回
~~¥28,400~~	~~¥15,100~~	~~¥10,400~~	~~¥8,200~~

X68000 Compact 大好評記念プレゼント!!

—あなたのオクトから素敵な贈物—

今、Compactをお買い上げいただいた方は、プレゼントの①番か②番のどちらかをお選び下さい。プラス③番は、もれなくプレゼント!!

①

※ソフトは変更出来ます!

or

②

③ ●MF-2HD（5枚）
●シリコンキーボードカバー
もれなく!! サービス!!

※どちらかお選び下さい!!
（どっちが得かヨーク考えてネ!）

ハードディスク（送料¥1,000）

■システムサコム SCSI
- ●HD-J040 ¥89,000 42M/25ms……… 大特価¥ **61,000**
- ●HD-J100 ¥128,000 100M/20ms……… 大特価¥ **83,000**
- ●HD-J130 ¥148,000 130M/20ms……… 大特価¥ **99,000**
- ●HD-J170 ¥189,800 173M/20ms……… 大特価¥**116,000**

■ロジテック SCSI
- ●LHD-FM100E ¥99,800 100M……… 大特価¥ **69,000**
- ●LHD-FM200E ¥138,000 200M/17ms……… 大特価¥ **95,000**

■エニックス SCSI
- ●EFX-100B ¥118,000 100M/19ms…… 大特価TEL下さい
- ●EFX-140B ¥138,000 140M/16ms…… 大特価TEL下さい

周辺機器コーナー（送料¥500）

品名	定価	特価
●CZ-6BE2A 2MB RAM（CZ-634C/644C用）	（¥59,800）	▶特価**¥42,500**
●CZ-6BE2B 2MB RAM（CZ-634C/644C/674C用）	（¥54,800）	▶特価**¥39,200**
●CZ-6BE2D 2MB RAM（CZ-674C用）	（¥54,800）	▶特価**¥39,000**
●CZ-6BE2 2MB RAM	（¥79,800）	▶特価**¥59,000**
●CZ-6BE4C 4MB RAM	（¥98,000）	▶特価**¥73,000**
●CZ-6BFI 増設用RS-232ボード	（¥49,800）	▶特価**¥35,800**
●CZ-6BGI GP-IBボード	（¥59,800）	▶特価**¥42,800**
●CZ-6BMI MIDIボード	（¥26,800）	▶特価**¥19,200**
●CZ-6BNI スキャナ用パラレルボード	（¥29,800）	▶特価**¥21,500**
●CZ-6BPI 数値演算プロセッサボード	（¥79,800）	▶特価**¥57,000**
●CZ-6BOI ユニバーサルI/Oボード	（¥39,800）	▶特価**¥29,800**
●CZ-6CRI CZ-6TU用RGBケーブル	（¥4,500）	▶特価**¥3,400**
●CZ-6VTI/BK カラーイメージ・ユニット	（¥69,800）	▶特価**¥52,000**
●CZ-8NM2A マウス	（¥6,800）	▶特価**¥5,100**
●CZ-8NTI マウストラックボール	（¥9,800）	▶特価**¥7,300**
●CZ-8NSI カラーイメージスキャナ	（¥188,000）	▶特価**¥132,000**
●CZ-6BCI FAXボード	（¥79,800）	▶特価**¥57,000**
●CZ-8TM2 モデムユニット	（¥49,800）	▶特価**¥37,000**
●LC-10CI-H カラー液晶ディスプレイ	（¥59,800）	▶特価**¥45,800**
●CZ-6TU GY/BK RGBシステムチューナー	（¥33,100）	▶特価**¥23,800**
●BF-68PRO 高性能CRTフィルター	（¥19,800）	▶特価**¥14,500**
●CZ-6MOI 光磁気ディスクユニット	（¥450,000）	▶特価**¥330,000**
●CZ-6BSI SCSIインターフェースボード	（¥29,800）	▶特価**¥22,000**
●CZ-6BL2 LANボード	（¥298,800）	▶特価**¥219,000**
●CZ-6BVI（ビデオボード）	（¥21,000）	▶特価**¥15,400**
●CZ-6BP2 数値演算プロセッサ	（¥45,800）	▶特価**¥34,300**
●AN-S100 スピーカーシステム（2本1組）	（¥36,600）	▶特価**¥26,300**
●JX-220X カラーイメージスキャナー	（¥168,000）	▶特価**¥120,000**
●CZ-6CSI SCSI変換ケーブル	（¥12,000）	▶特価**¥8,900**
●CZ-6CTI CZ-6TU用テレビコントロールケーブル	（¥5,500）	▶特価**¥4,200**

※クレジットの回数は1回〜60回、ボーナス併用などありますのでお電話でお問合せ下さい。

■本体セット:送料無料（注）本体セット以外の周辺機器（プリンター、モデム、HDD等）及びソフトの送料は、北海道・九州地区＝1ケロ¥1500、■その他離島地区は、1ケロ¥2000となります。

※上記料金には、消費税は含まれておりません。消費税が付加されますので、詳しくは、電話でお問合せ下さい。

■便利です。夜9時まで営業しております。お立寄り下さい。お待ちしております!!

■特に人気のある商品によっては、しばらくお待ち願うことがありますのでご了承下さい!!

■ビッグバーゲンセール実施中!!ゲームソフト(ビジネス)新製品続々入荷中!!

X68000 ——堂々のラインアップ!!——

■PROII/XVI■

ボーナス〈2回・4回・6回〉払いOK!!手数料無料!!

X68000 XVI エクシヴィ

(送料・消費税込)
超特価¥残念!表示不能!!

X68000XVI ドッカーン!プレゼント!!
—あなたのオクトから素敵な贈物—

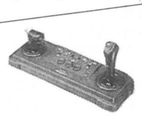
今、XVIをお買い上げいただいた方は、プレゼントの①番か②番のどちらかお選び下さい。プラス③番はもれなくプレゼント!!

① F15ストライクイーグルII (定価¥10,800)

新作!大人気ゲームソフト!!
銀河英雄伝説II デラックスセット (定価¥12,600)

or

※どちらかお選び下さい!!

② インテリジェントコントローラ
■CZ-8NJ2
(CYBER STICK)
シューティングゲーマーの必須アイテム!!
(定価¥23,800)


\+

③ MD-2HD(10枚)
シリコンキーボードカバー
もれなく!!サービス!!

■CZ-634C-TN (定価¥368,000)

Ⓐ●CZ-634C-TN+CZ-606D-TN
定価合計¥447,800▶**超特価¥表示不能!**

12回	~~¥27,600~~	24回	~~¥14,600~~	36回	~~¥10,100~~	48回	~~¥ 8,000~~

Ⓑ●CZ-634C-TN+CZ-614D-TN
定価合計¥503,000▶**超特価¥表示不能!**

12回	~~¥31,200~~	24回	~~¥16,600~~	36回	~~¥11,500~~	48回	~~¥ 9,000~~

■CZ-644C-TN (定価¥518,000)

Ⓒ●CZ-644C-TN+CZ-606D-TN
定価合計¥597,800▶**超特価¥表示不能!**

12回	~~¥37,400~~	24回	~~¥19,800~~	36回	~~¥13,700~~	48回	~~¥10,800~~

Ⓓ●CZ-644C-TN+CZ-614D-TN
定価合計¥653,000▶**超特価¥表示不能!**

12回	~~¥40,900~~	24回	~~¥21,700~~	36回	~~¥15,000~~	48回	~~¥11,800~~

※クレジット表は、送料・消費税込!

X68000PROII ラストチャンス!! 〈BIGプレゼント付〉 (送料無料・税別)

X68000PROII
(CZ-653C)

定価¥285,000
超特価~~¥138,000~~

新作!大人気ゲームソフト!!
銀河英雄伝説II デラックスセット

(定価¥10,800)
プレゼント
さらに! ★JOY CARD(連射式)×2個
さらにさらに!! ★MD-2HD 10枚

■CZ-653C (定価¥285,000)

Ⓐ●CZ-653C+CU-21HD
定価合計¥433,000▶**超特価¥239,000**

12回	¥22,100	24回	¥11,700	36回	¥ 8,100	48回	¥ 6,400

Ⓑ●CZ-653C+CZ-606D
定価合計¥364,800▶**超特価¥195,000**

12回	¥18,100	24回	¥ 9,600	36回	¥ 6,600	48回	¥ 5,200

Ⓒ●CZ-653C+CZ-607D
定価合計¥384,800▶**超特価¥209,000**

12回	¥19,400	24回	¥10,300	36回	¥ 7,100	48回	¥ 5,600

Ⓓ●CZ-653C+CZ-614D
定価合計¥420,000▶**超特価¥229,000**

12回	¥21,200	24回	¥11,200	36回	¥ 7,800	48回	¥ 6,100

X68000ソフト大セール実施中!! (ゲームソフト25〜30%OFF) (送料¥500)

〈グラフィック〉●Z's STAFF PRO68K Ver.2.0 (シャフト)定価¥58,000 ……特価**¥36,500**

〈開発ツール〉●CコンパイラPRO-68K Ver.2.1 定価¥44,800 CZ-285LSD ……特価**¥32,500**

〈統合表計算ソフト〉BUSINESS PRO-68K Popular 定価¥28,000 CZ-286BSD ……特価**¥21,000**

〈レイアウト〉●Pressconductor PRO-68K 定価¥28,000 CZ-268BSD ……特価**¥21,000**

〈C言語〉●C & Professional Pack 定価¥58,000 ……特価**¥39,600**

〈音楽〉●Music studio PRO-68K Ver. 2.0 定価¥28,800 CZ-261MS ……特価**¥21,200**

〈CGシール〉●CANVAS PRO-68K 定価¥29,800 CZ-249GS ……特価**¥22,200**

〈ワープロ〉●Multiword Ver.1.1 定価¥32,000 CZ-225BSD ……特価**¥23,000**

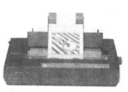
〈OS〉●OS-9 X68000 Ver.2.4 定価¥35,800 CZ-284SSD ……特価**¥26,900**

型名	商品	定価	特価
CZ-212BS	〈BUSINESS PRO-68K〉	(¥ 68,000)	¥ 48,000
CZ-213MS	〈MUSIC PRO68K〉	(¥ 18,800)	¥ 13,200
CZ-275MWD	〈SOUND SX-68K〉	(¥)	¥TEL下さい
CZ-215MS	〈Sampling PRO-68K〉	(¥ 17,800)	¥ 12,500
CZ-287SS	〈SX-WINDOW Ver.2.0〉	(¥ 12,800)	¥ 9,600
CZ-220BS	〈DATA PRO-68K〉	(¥ 58,000)	¥ 40,000
CZ-272CWD	〈Communication SX-68K〉	(¥ 19,800)	¥ 15,300
CZ-224LS	〈THE 福袋 V2.0〉	(¥ 9,900)	¥ 7,400
CZ-253BS	〈CARD PRO-68K Ver.2.0〉	(¥ 29,800)	¥ 20,800
CZ-258BS	〈Tlepotion PRO-68K〉	(¥ 22,800)	¥ 16,800
CZ-244SS	〈Homan 68K Ver.2.0〉	(¥ 9,800)	¥ 7,500
CZ-247MS	〈MUSIC PRO-68K〔MIDI〕〉	(¥ 28,800)	¥ 20,800
CZ-240BS	〈Stationery PRO-68K〉	(¥ 14,800)	¥ 11,500
CZ-243BS	〈CYBER NOTE PRO-68K〉	(¥ 19,800)	¥ 15,200

型名	商品	定価	特価
	〈Z's TRIPHY(デジタルクラフト)〉	(¥ 39,800)	¥ 27,000
	〈テラッツオ(ハミングバード)〉	(¥ 19,400)	¥ 13,600
	〈KAMIKAZE(サムシンググッド)〉	(¥ 68,000)	¥ 43,800
	〈Final Ver.3.2(エーエスピー)〉	(¥ 38,000)	¥ 29,000
	〈サイクロンEXPRESSα68〉	(¥ 98,000)	¥ 69,000
	〈Gツール(ザインソフト)〉	(¥ 28,000)	¥ 18,600
	〈たーみのる2(SPS)〉	(¥ 17,800)	¥ 13,000
	〈G68K Ver.2 PRO〉	(¥ 22,000)	¥ 17,300
	〈C-TRACE68 Ver.3.0〉	(¥ 98,000)	¥ 68,500
CZ-251BS	〈ハイパーワード〉	(¥ 39,800)	¥ 29,400
CZ-260LS	〈XBAS to CHECKER PRO-68K〉	(¥ 9.800)	¥ 7,500
CZ-234LS	〈AI-68K〉	(¥188,000)	¥139,000
CZ-255GS	〈CANVASドローグラフィックLIB〉	(¥ 8,800)	¥ 6,600
CZ-256GS	〈CANVASドローグラフィックVol.2〉	(¥ 8,800)	¥ 6,600

プリンタ (送料¥1,000)

■CZ-8PC5-BK
熱転写カラー漢字
定価¥96,800
大特価**¥68,800**

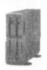


■IO-735X-B
カラーイメージジェット
定価¥248,000
大特価**¥154,000**

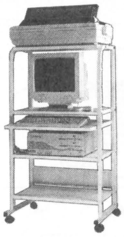
今月の推奨品 (送料¥1,000)

■内蔵用ハードディスク
〈Compact XVI(CZ-674C)用〉
[KGU-HD80K]
Compact HD-80キット
定価¥168,000

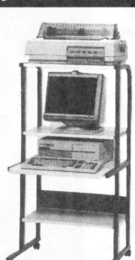
限定特別価格¥TEL下さい!

■5インチフロッピーディスクユニット
〈X68000Compact(CZ-674C-H)用〉
[CZ-6FD5]

定価¥99,800
限定特別価格¥TEL下さい!

パソコンラック〈送料無料〉



Ⓐ5段キャスター付
スライド式キーボード台
●1150(H)×640(W)×600(D)
定価¥38,000
特価
¥12,500



Ⓑ4段キャスター付
●1250(H)×640(W)×700(D)
定価¥29,800
特価
¥8,800

店頭新作ゲームソフト25〜30%OFF!!ビジネスソフト25%より特価中

★通信販売お申込みのご案内★ 〒144 東京都大田区蒲田4-6-7 TEL:03-3730-6271

お申込みはお電話でお願いします。お客様の〈住所〉〈氏名〉〈電話番号〉及び〈商品名〉をお知らせ下さい。●入金確認後ただちに商品をご送付いたします。

現金一括払い
銀行振込:お近くの銀行より(電信扱い)にてお振込み下さい。
現金書留:封筒の中に住所・氏名・商品名をご記入の上当社までお送り下さい。

クレジット
専用お申込用紙をお送り致します。ので、必要事項をご記入、ご捺印の上ご返送下さい。手続きは簡単です。

オクト ラクラク クレジット表

3回	3.5	6回	4.5	10回	6.0	12回	6.0
15回	9.0	18回	11.0	20回	12.0	24回	12.5
30回	17.0	36回	17.5	48回	23.0	60回	33.0

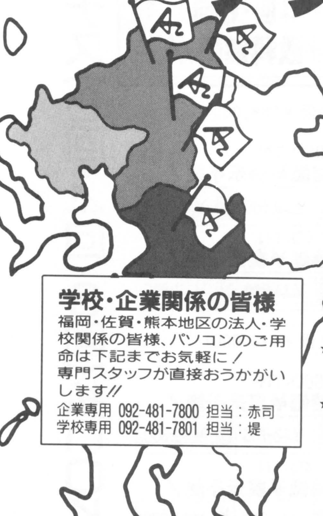
振込先
富士銀行 久ヶ原(クガハラ)支店 ㊟No.1824
三菱銀行 蒲田支店 ㊟No.0278691
株式会社 億人(オクト)

※掲載の価格は変動しますので、まずは、お電話にてご確認ください。
※上記料金には、消費税は含まれておりません。消費税が付加されますので、詳しくは電話でお問合せ下さい。
※銀行振込、または、現金書留でご注文の際には、あらかじめ電話でご確認の上、お申し込み下さい。

■ボーナス〈2回・4回・6回〉払いOK!!手数料無料!!ご利用下さい。■店頭にて、新作ゲームソフト25〜30%OFF!!

九州のSUPER SHOP

アプライドの超高額買取りバリ高か～～！



SUPER SHOPはここが違う!!

★商品到着後翌日までに、査定連絡！即入金！バリ早か～～！
★パソコン以外でも何でも買取れます。！バリ高か～～！

アプライドが買取れるもの：パソコン・パソコン周辺機器・ワープロ・ファミコン・ゲームソフト・オーディオ・カメラ・電子楽器（キーボードなど）・ビデオ・CD・LD・レコード・図書券・商品券など・古銭・切手・古本などなどな～んでも高く買います!!

一度博多に行ってみるべし!! 送ってみるべし!!

★印商品 高額 買取り中!!

学校・企業関係の皆様
福岡・佐賀・熊本地区の法人・学校関係の皆様、パソコンのご用命は下記までお気軽に！
専門スタッフが直接おうかがいします!!
企業専用 092-481-7800 担当：赤司
学校専用 092-481-7801 担当：堤

〈買取り価格一覧〉

市場相場に伴い買取り価格は常に変動します。(8/1現在)
この買取り価格は、システムディスク・マニュアル・箱など全て揃った価格です。

パソコン本体	
★NEC PC-9801FA2	→¥ 230,000
NEC PC-9801FS	→¥ 182,000
NEC PC-9801FX	→¥ 153,000
★NEC PC-9801DA2	→¥ 195,000
NEC PC-9801RA21	→¥ 165,000
NEC PC-9801RA2	→¥ 145,000
NEC PC-9801RS21	→¥ 120,000
NEC PC-9801DX2	→¥ 120,000
NEC PC-9801RX21	→¥ 105,000
NEC PC-9801RX2	→¥ 95,000
NEC PC-9801VX21	→¥ 75,000
NEC PC-9801DO+	→¥ 70,000
NEC PC-9801DO	→¥ 60,000
NEC PC-9801CV21	→¥ 65,000
NEC PC-9801UV11	→¥ 55,000
NEC PC-9801UV21	→¥ 45,000
NEC PC-9801UV2	→¥ 40,000
NEC PC-9801LV22	→¥ 35,000
NEC PC-9801LV21	→¥ 20,000
NEC PC-9801NC	→¥ 250,000
NEC PC-9801NS20	→¥ 100,000
NEC PC-9801NS	→¥ 60,000
NEC PC-9801NV	→¥ 45,000
NEC PC-9801N	→¥ 40,000
★NEC PC-9801NSE	→¥ 130,000
★NEC PC-9801NST	→¥ 150,000
EPSON PC-486GR2/5	→¥ 252,000
EPSON PC-486GF2/5	→¥ 192,000
EPSON PC-386S	→¥ 170,000
EPSON PC-386GS2	→¥ 185,000
EPSON PC-386P	→¥ 120,000
EPSON PC-386GE2	→¥ 130,000
EPSON PC-386V-STD	→¥ 130,000
EPSON PC-386M-STD	→¥ 120,000
EPSON PC-386VR-STD	→¥ 105,000
EPSON PC-386LSR	→¥ 135,000
EPSON PC-386LS-STD	→¥ 80,000
EPSON PC-286VX-STD	→¥ 80,000
EPSON PC-286VG-STD	→¥ 80,000
EPSON PC-286VF-STD	→¥ 60,000
EPSON PC-286VS-STD	→¥ 65,000
EPSON PC-286UX-STD	→¥ 60,000
EPSON PC 286US-STD	→¥ 45,000
EPSON PC 286LP-STD	→¥ 80,000
EPSON PC 286LST	→¥ 110,000
EPSON PC 286LS-STD	→¥ 90,000
EPSON PC 286LE-STD	→¥ 50,000
EPSON PC 286LF-STD	→¥ 65,000
EPSON PC 286L-STD	→¥ 30,000
EPSON PC 286BOOKL	→¥ 155,000
EPSON PC 286BOOKS	→¥ 60,000
EPSON PC 386NOTEW	→¥ 100,000
★EPSON PC 386NOTEA	→¥ 120,000
EPSON PC 286NOTEF	→¥ 40,000
EPSON PC CLUB	→¥ 50,000
SHARP X68 SUPER HD	→¥ 145,000
SHARP X68 EXPERT2 HD	→¥ 115,000
SHARP X68 EXPERT	→¥ 85,000
SHARP X68 PRO2 HD	→¥ 95,000
SHARP X68 PRO HD	→¥ 90,000
SHARP X68 PRO	→¥ 80,000
SHARP X68 ACE HD	→¥ 75,000
SHARP X68 ACE	→¥ 65,000
FUJITU TOWNS80H+DP531	→¥ 170,000
FUJITU TOWNS40H+DP531	→¥ 140,000
FUJITU TOWNS20F+DP531	→¥ 100,000
FUJITU TOWNS2F+DP531	→¥ 70,000
APPLE MAC II si(5M.40)Set	→¥ 200,000
APPLE MAC CLASIC	→¥ 40,000
APPLE MAC CLASIC(2M.45)	→¥ 55,000

プリンタ	
NEC PC-201X	→¥ 50,000
NEC PC-PR201J	→¥ 40,000
NEC PC-PR201B	→¥ 20,000
NEC PC-PR201V2	→¥ 30,000
NEC PC-PR201 80A	→¥ 80,000
NEC PC-PR201H3	→¥ 28,000
NEC PC-PR201G	→¥ 30,000
NEC PC-PR150VH	→¥ 30,000
NEC PC-PR150VL	→¥ 25,000
NEC NM9950 2	→¥ 20,000
NEC NM9950	→¥ 10,000
NEC PC-PR101G2	→¥ 20,000
NEC NM-4150	→¥ 50,000
EPSON VP-4800	→¥ 50,000
EPSON VP-135EX	→¥ 15,000
EPSON VP-1600	→¥ 35,000
EPSON VP-1500	→¥ 28,000
EPSON AP-800	→¥ 20,000
EPSON AP-850→¥	→¥ 25,000

中古本体	
スーパーファミコン	→¥ TEL
ファミコン	→¥ 4,000
ゲームボーイ	→¥ 3,000
ディスクシステム	→¥ 1,300
ツインファミコン	→¥ 8,000
PCエンジン(白)	→¥ 4,000
〃 (シャトル)	→¥ 3,500
〃 (コアグラ)	→¥ 4,500
〃 (コアグラII)	→¥ 5,000
〃 (スーパーグラフィックス)	→¥ 4,500
〃 (GT)	→¥ 12,000
CD-ROM2 Ver1.0	→¥ 11,000
〃 Ver2.1	→¥ 16,000
PCエンジンDOO	→¥ 22,000
メガドライブ	→¥ 4,000
メガCD	→¥ 16,000
ゲームボーイ	→¥ 4,000
ゲームギア	→¥ 4,000
ネオ・ジオ	→¥ 17,000

サプライ	
GGワイドギア	→¥ 400
GTTVチューナ	→¥ 3,800
MDアーケードパワースティク	→¥ 2,300
MDコントロールパッド	→¥ 600
MDメガアダプター	→¥ 800
FCジョイカードMK2	→¥ 300
FCバックスパワーグローブ	→¥ 300
GBサウンドボーイ	→¥ 500
GB充電式ACアダプタ	→¥ 1,600
PC XE-IPROHE	→¥ 3,800
PC ターボパットグレー	→¥ 600
PC 天の声2	→¥ 1,600
PC バックアップブースター I	→¥ 3,300
SF JBキング	→¥ 2,800
SF ステレオAVケーブル	→¥ 400

攻略本	
定価300円～499円まで	→¥ 100
定価500円～999円まで	→¥ 200
定価1,000円～1,499円まで	→¥ 300
定価1,500円～	→¥ 400

ファミコン

赤川次郎の幽霊列車
アフターバーナー
イース3
囲碁指南
囲碁名鑑
いただきストリート
ウィザードリィ
A列車で行こう
絵描衛門
ゴルフグランドスラム
ぎゅわんぶらぁ自己中心派2
快傑ヤンチャ丸2
勝馬伝説
川の主釣り
がんばれゴエモン2
SD刑事ブレイダー
オセロ
おたくの星座
クォーターバックスクランブル
くにおくんの時代劇だよ全員集合
甲子園
五目ならべ

買取り急上昇!!
※ソフトの買取り価格は、お電話にてお問い合せ下さい。

X68000 コーナー

モニター	
CZ-600D	→¥27,000
CZ-602D	→¥28,000
CZ-604D	→¥25,000
CZ-605D	→¥30,000
CZ-606D	→¥20,000
CZ-607D	→¥30,000
CZ-608D	→¥27,000
CZ-611D	→¥33,000
CZ-612D	→¥33,000
CZ-613D	→¥42,000
CZ-614D	→¥42,000
CU-21CD	→¥41,000
CU-21HD	→¥44,000

本体	
CZ-600C	→¥ 55,000
CZ-601C(ACE)	→¥ 60,000
CZ-611C(ACE/HD)	→¥ 80,000
CZ-652C(PRO)	→¥ 75,000
CZ-662C(PRO/HD)	→¥ 95,000
CZ-653C(PRO II)	→¥ 90,000
CZ-663C(PRO II/HD)	→¥110,000
CZ-602C(EXPERT)	→¥ 80,000
CZ-612C(EXPERT/HD)	→¥105,000
CZ-603C(EXPERT II)	→¥ 95,000
CZ-613C(EXPERT II/HD)	→¥115,000
CZ-623C(SUPER/HD)	→¥145,000
CZ-604C(SUPER)	→¥120,000
CZ-674C(Compact XVI)	→¥145,000
CZ-644C(XVI/HD)	→¥185,000
CZ-634C(XVI)	→¥155,000

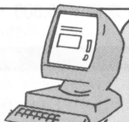
アプライドの24時間テレホン情報
092-481-9964

APPLIED買取りセンター直通
0120 FreeDial **0120-488998**
買い取り専用FAX **092-481-6143**
〈郵送の方はこちらへ！〉

通信販売用お振込み先
西日本銀行赤坂門支店
普No.0915753 アプライド㈱

アプライドの買取方法は3パターンあるんだ！

発送の場合

宅配便などでアプライドにお送り下さい

商品到着後翌日までには査定価格の連絡をします
お支払い方法は
・現金書留
・口座振込み
のどちらかです
振込希望の方は口座番号と名前のフリガナをお忘れなく

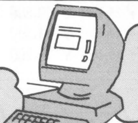
直接お店に持込みの場合

アルスターくん

商品と身分証明書と印鑑をお持ちください

ぼくが、その場で査定します。
よろしく!!

出張買取の場合
FreeDial 0120-488998
にお電話下さい!!
お客様の都合に合せます！

指定の時間に車でご自宅までうかがいます。

その場で査定 即金でお支払いします!!
出張買取りは福岡市周辺のみです

アプライド博多店
担当：室園

大駐車場完備
〒812 福岡市博多区豊2丁目3－10
TEL(092)481-7800・FAX(092)481-7651
年中無休/営業時間AM9:00～PM8:00

アプライド福岡店
担当：松村

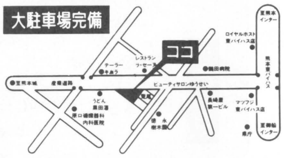
駐車場完備
〒810 福岡市中央区大手門1-1-1 平和台ビル1F
TEL(092)712-8099・FAX(092)751-2112
営業時間（平日/AM10:30～PM8:00 日・祭/AM10:00～PM8:00）

アプライド大橋店
担当：高本

駐車場完備
〒815 福岡市南区向野2丁目12－30
TEL(092)542-1155・FAX(092)542-1156
営業時間（月～木/AM11:00～PM8:00 金・土/AM11:00～PM9:00 日・祭/AM10:00～PM8:00）

アプライド久留米店
担当：島原

大駐車場完備
〒830 久留米市東櫛原町字渕の上町293－1
TEL(0942)33-7968・FAX(0942)33-8285
年中無休/営業時間AM9:00～PM8:00

アプライド熊本店
担当：新
大駐車場完備

〒862 熊本市保田窪2丁目1－26
TEL(096)384-0901 FAX(096)384-8881
年中無休/営業時間AM10:00～PM10:00

新品・中古品のお求めは九州をネットするアプライド各店へどうぞ!!（パソコン・ファミコン何でもOK）

X68000 Pro SHOP
BASIC HOUSE
KEISOKUGIKEN Corp.
TEL 0286-22-9811　FAX 0286-25-3970

絶賛発売中！
CD-ROM Drive for X68000
マルチメディアへの誘い

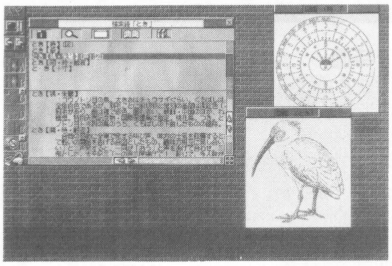
FirstClassTechnology制作のCD-ROM Device Driverを付属させ、ついにX68000用CD-ROM Driveの登場です。本製品を使用することにより、MS-DOSやPC-9801シリーズ、FM-TOWNSなどで採用されている、ISO9660規格のCDをHuman68K/SX-WINDOWで直接扱えるようになります。

また、将来の拡張にも柔軟に対応できるSCSIインターフェースによる接続を採用。ディジーチェーンによって既存のSCSIハードディスクとの同時使用も可能です。

担当　登坂高明

9月発売

CD-ROM広辞苑 検索&閲覧
SX-WINDOW



富士通・NECより発売されている12cmCD-ROM広辞苑を検索/参照し、X68000上で使用できるようにする専用検索ソフトです。通常の検索に加えて、文章中語検索、条件検索などの強力な検索機能が利用でき、広辞苑内に収録されている図版の表示、音声の再生も可能です。SX-WINDOW上で動作するので、SX-WINDOWの特徴である疑似マルチタスク機能やカット＆ペースト等の機能が利用でき、エディタXなどで文章編集中に検索/参照することが簡単にできます。また将来ワープロなどのSX-WINDOW対応ソフトが発売された場合にも本ソフトの使用によって広辞苑の有効利用が可能になります。同時に複数起動ができるため、いくつかの情報を同時に参照しながら作業を進めることも可能です。

標準価格　検索ソフトのみ　¥19,800-
　　　　　CD広辞苑セット　¥47,800-

※本アプリケーションの実行にはメインメモリ2Mバイト以上実装している必要があります。

ドライブ仕様

型番	KGU-XCD
使用ドライブ	東芝　XM-3301
平均アクセスタイム	325mSEC
インターフェース	SCSI
キャッシュメモリー	64KB
オーデオ出力	RCA-Phono端子×2 ステレオヘッドホン端子
電源	専用ACアダプター
外形寸法	150×228×50（電源部含まず）

※SCSIケーブル・ターミネーターは別売になります。

付属サポートソフト

ISO9660準拠デバイスドライバ
MusicPlayer for SX-Window
Macintosh™用ファイルビューア　for SX-Window

KGU-XCD対応
X68000 CD-ROM第一弾！「フリーウェア集」
Free Soft Ware Selection - CD68K
近日発売

標準価格¥118,000-

※Macintosh™はAppleComputerの登録商標です
※特価セットは、通販のみです。

低金利クレジット　通信販売送料 全国一律¥1,000 長期クレジット可能
※表示価格に消費税は含まれておりません
株式会社 計測技研　マイコンショップ BASIC HOUSE 本社／ショールーム／通販部
〒321 栃木県宇都宮市竹林町503-1
TEL 0286-22-9811　FAX 0286-25-3970

OS-9 Ver.2.4 対応パッケージ リリース

microware

OS-9/X68000 C＆Professional Pack. V3.2

C＆プロフェッショナル・パッケージは、OS-9/X68000上で動作するマイクロウェア・Cコンパイラとユーティリティ・ソフトのパッケージです。

◆マイクロウェア・Cコンパイラの特徴

他OSの、K&R準拠、ANSI準拠のアプリケーション、あるいはUNIX上のアプリケーションは、特に修正することなく容易に移植できます。

このコンパイラはCPUのインストラクションに最適化されており、生成されるオブジェクトが、最小、最速になるように複数レベルに渡るオプティマイズを実行しています。

◆拡張機能

1. シンボリック・デバッグ
2. 強力なエラー診断機能
3. 高速なコンパイル・スピード
4. 豊富なオプション
5. アセンブリ言語とのインタフェース
6. OS-9/X68000用拡張ライブラリ

※バージョンアップサービスを予定しておりますので、お早めにユーザ登録をお済ませ下さい。

◆付属ユーティリティ・ソフト

●SrcDbg(ソース・レベル・デバッガ)
SrcDbgは、C言語で書かれたプログラムのテストやデバッグをソース・レベルで行うユーティリティです。

●μMACS(マイクロマックス)
μMACSは、UNIX上で広く利用されているスクリーンエディタ"EMACS"のOS-9版サブセットです。

◆パッケージ内容
マイクロウェア・Cコンパイラ
標準ライブラリ
OS-9/X68000専用ライブラリ
ヘッダ・ファイル
OS-9/X68000専用ヘッダ・ファイル
アセンブラ
リンカ
ユーザステート・シンボリック・デバッガ
ソース・レベル・デバッガ
漢字フル・スクリーン・エディタ

◆付属マニュアル
Cコンパイラ・ユーザーズ・マニュアル
アセンブラ・リンカ・デバッガ・ユーザーズ・マニュアル
ソース・レベル・デバッガ・ユーザーズ・マニュアル
μMACSユーザーズ・マニュアル
OS-9/X68000専用ライブラリ・マニュアル1,2

3.5"2HD　5"2HD　2枚組
定価￥80,000

OS-9/X68000テクニカル・デベロップメント・キット Technical Development Kit V2.4

OS-9/X68000テクニカル・デベロップメント・キットには、OS-9上でのプログラミングのためのマニュアルとシステム・ステートでのデバッグを可能とするデバッガが含まれています。

また、デバイス・ドライバ作成のために、※各種サンプルソースコードが付属しています。

※サンプルソースコードに関してのお問い合わせはご遠慮願います。

◆パッケージ内容
マニュアル
システムコール
テクニカル
I/Oテクニカル
システムステート・デバッガ・ユーザーズ
ROMデバッガ・ユーザーズ
ソフトウェア
SysDbg
RomBUG
※各種サンプル・ソースコード

◆システムステート・デバッガ(SysDbg)
SysDbgは、OS-9システムの拡張など、I/Oドライバの開発を強力に支援するシンボリック・ステート・デバッガです。

◆ROMデバッガ(RomBUG)
RomBUGは、OS-9とは独立したデバッガです。起動時に必要なコンソールやディスクなどのデバイス・ドライバをデバッグすることができます。

供給メディア
3.5"2HD 5"2HD
定価　￥38,000円

＊会社名・製品名は、各社の商標または登録商標です。
※製品の内容等は予告なく変更されることがあります。

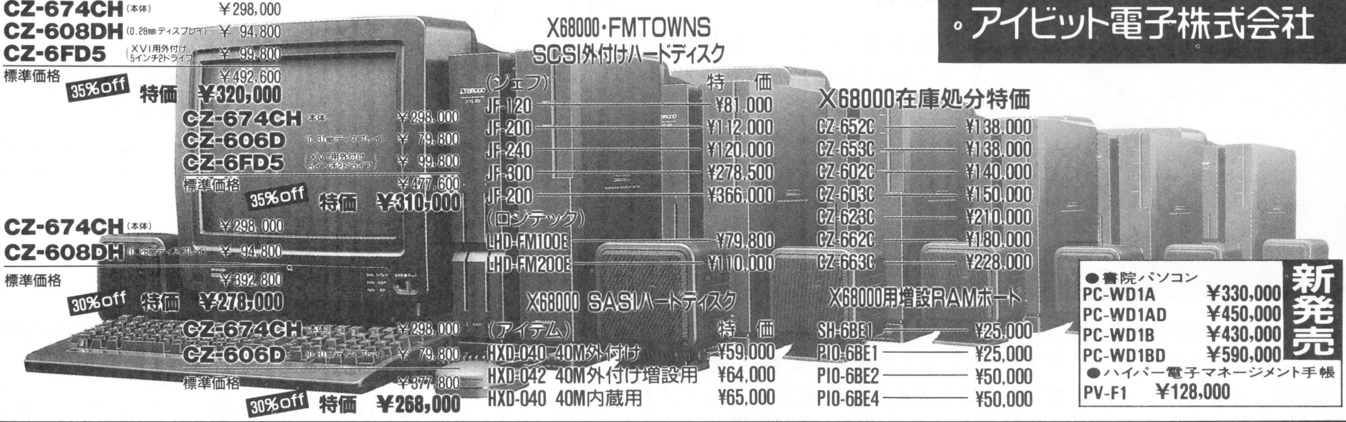
OS-9/X68000はシャープ㈱から販売しています。

マイクロウェア・システムズ株式会社
〒101 東京都千代田区外神田2-17-3　代表(03)3257-9000　Fax(03)3257-9200

X68000シリーズ 大特価セール！

9月末迄

AIBIT
アイビット電子株式会社



CZ-674CH (本体) ￥298,000
CZ-608DH (0.28mmディスプレイ) ￥94,800
CZ-6FD5 (XVI用外付け 5インチ2ドライブ) ￥99,800
標準価格 ￥492,600
35%off 特価 ￥320,000

CZ-674CH (本体) ￥298,000
CZ-606D (0.31mmディスプレイ) ￥79,800
CZ-6FD5 (XVI用外付け 5インチ2ドライブ) ￥99,800
標準価格 ￥477,600
35%off 特価 ￥310,000

CZ-674CH (本体) ￥298,000
CZ-608DH (0.28mmディスプレイ) ￥94,800
標準価格 ￥392,800
30%off 特価 ￥278,000

CZ-674CH (本体) ￥298,000
CZ-606D (0.31mmディスプレイ) ￥79,800
標準価格 ￥377,800
30%off 特価 ￥268,000

X68000・FMTOWNS SCSI外付けハードディスク

	特価
(ジェフ)	
JF-120	￥81,000
JF-200	￥112,000
JF-240	￥120,000
JF-300	￥278,500
JF-200	￥366,000
(ロジテック)	
LHD-FM100E	￥79,800
LHD-FM200E	￥110,000

X68000 SASIハードディスク

(アイテム)	特価
HXD-040 40M外付け	￥59,000
HXD-042 40M外付け増設用	￥64,000
HXD-040 40M内蔵用	￥65,000

X68000在庫処分特価

CZ-652C	￥138,000
CZ-653C	￥138,000
CZ-602C	￥140,000
CZ-603C	￥150,000
CZ-623C	￥210,000
CZ-662C	￥180,000
CZ-663C	￥228,000

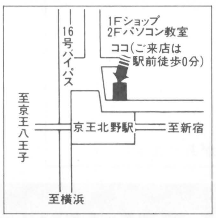
X68000用増設RAMボード

SH-6BE1	￥25,000
PIO-6BE1	￥25,000
PIO-6BE2	￥50,000
PIO-6BE4	￥50,000

新発売
●書院パソコン
PC-WD1A ￥330,000
PC-WD1AD ￥450,000
PC-WD1B ￥430,000
PC-WD1BD ￥590,000
●ハイパー電子マネージメント手帳
PV-F1 ￥128,000

他周辺機器及びポケコン全機種取り扱い。

〈全商品新品完全保証付〉
★シャープ・シャープ周辺機器(拡張機器全機種、プリンター他)・富士通・NEC常時取り扱い。
★シャープ・カシオポケコン全機種取り扱い。PACIFIC・YHP・キャノンも取り扱い。
★学校、企業納入受け賜ります。送料別料金。★上記商品価格には、消費税は含まれておりません。
★特価表及び資料をご希望の方は、72円切手を同封の上お送りください。
通信販売のお問い合せ、御注文は
TEL.0426-45-3001(本店) FAX.0426-44-6002
●営業時間/10:00～19:00●電話受付/9:00～22:00迄可●定休日/水曜日
SHARP SUPER EXE SHOP
アイビット電子株式会社　〒192 東京都八王子市北野町560-5



上記の広告商品はすべて店頭販売もしております。

全国通信販売
北海道から沖縄まで

★送料はご注文の際にお問い合わせ下さい。
★掲載の商品は、すべて新品、保証書付きです。
★掲載の商品は充分用意してありますが、ご注文の際は、在庫の確認の上、現金書留または、銀行振込でお申し込み下さい。全商品クレジットでも扱っております。
★お申し込みの際は必ず電話番号を明記して下さい。
★商品、品切れの節はご容赦下さい。

富士銀行八王子支店　(普)1752505

SHARP

提供するのは、X68000の才能をひき出す仕事です。

コンピューター事業拡張につき
プログラマー募集！

勤務地 大阪・東京・岡山

（男女不問・現地面接可）

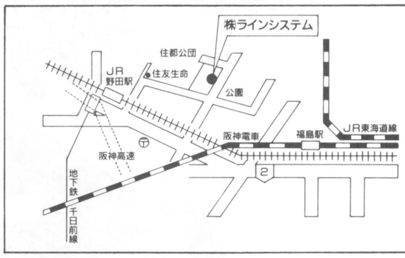
■会社概要
設　　立■昭和44年
資 本 金■1,500万円
従業員数■17名
平均年齢■26歳
■事業内容
パーソナルコンピュータ・AXによる自社ソフトパッケージの開発及びオーダーメイド販売サポート
X68000による画像作成業務
資　　格■高卒以上30歳位迄の方
※C言語、アッセンブラーの出来る方歓迎。未経験者も歓迎。
給　　与■経験・能力等与慮の上、当社規定により優遇いたします。例 25歳 ㊊ 176,000円
※別途報奨金制度あり
待　　遇■昇給年1回・賞与年2回 手当/業務・営業・皆勤 交通費全額支給
勤務時間■9:00〜18:00
福利厚生■各種社会保険完備 退職金制度 財形貯蓄制度 社内旅行有

経験の有無を問わず、X68000大好き人間 歓迎。経験者には、実力を発揮する場を、未経験者には丁寧な指導をお約束します。
シャープ、XEROX等のシステム機器販売から、シャープ・コンピューターのシステムプレゼンテーターとしてメーカーの期待を担う当社で活躍して下さい。

株式会社ラインシステム

本社　〒553 大阪市福島区鷺洲3丁目1 TEL06-458-7313　担当 菊田
〒115 東京都北区浮間3-2-16 エスポワール403 TEL03-5994-2087　担当 鈴木

休日休暇■隔週休2日制（完全週休2日制も検討中）
祝日
有給・特別・夏期・年末年始休暇等
応　　募■電話連絡の上、履歴書（写真貼付）を持参又は郵送して下さい。追って詳細を連絡いたします。
※入社日相談に応じます。
※応募の秘密厳守いたします。
交　　通■阪神、地下鉄野田駅下車 徒歩7分



GNU ツールボックス

UNIXからDOSへ──X68000の移植を通して

吉野智興 村上敬一郎　共著

B5変型判/240ページ　**定価2,200円**（税込）

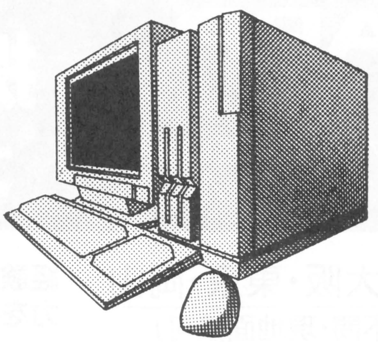
g++、gcc、Nemacsを、X68000に移植するその経緯とノウハウを紹介。68000系のマシンだけでなく、8086系のマシンへの移植も可能です。プログラムをUNIXからDOSへ移植しようとしている方に贈る一冊！

お近くの書店でお求め下さい
ソフトバンク出版事業部　SOFT BANK

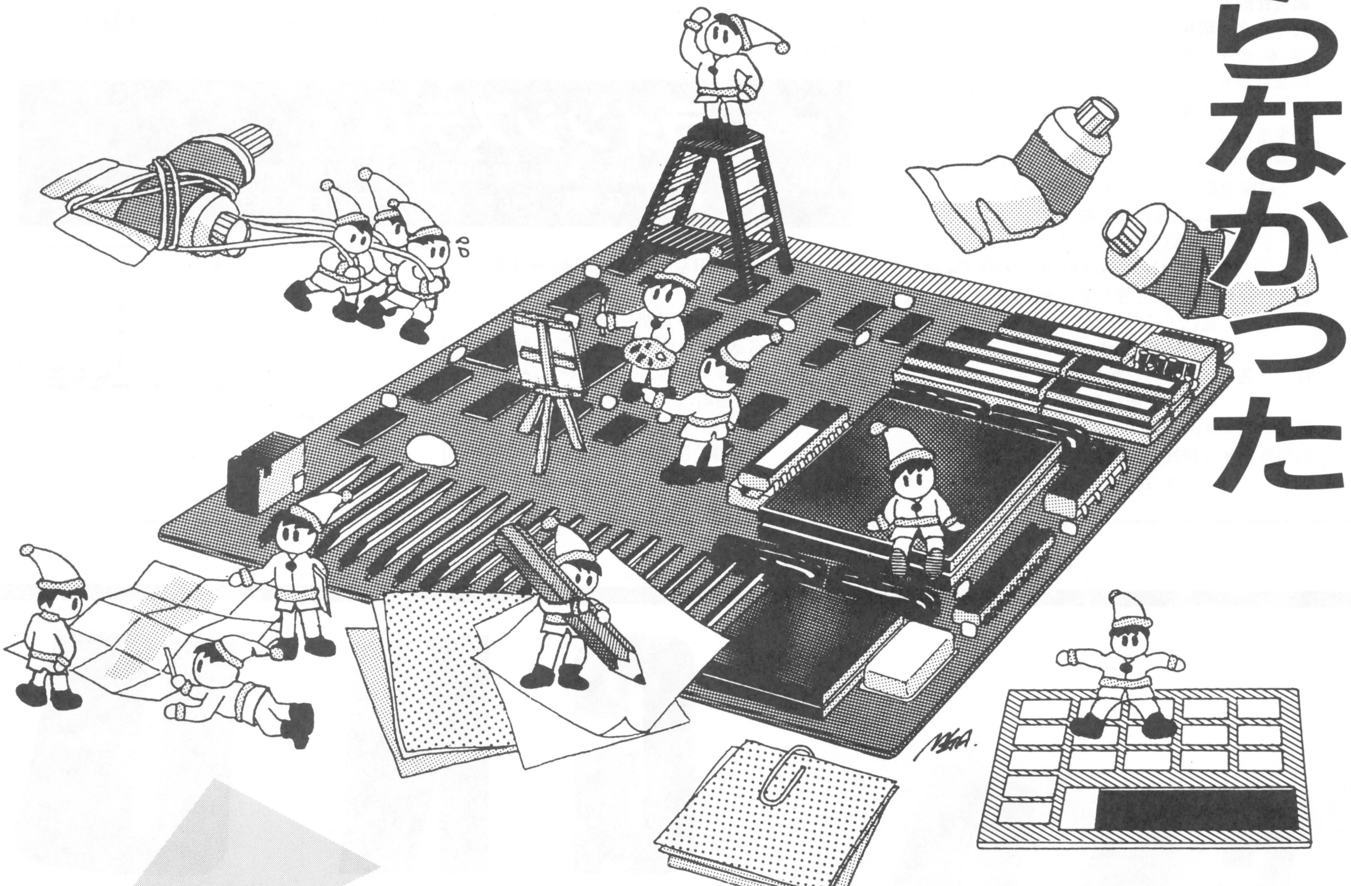
ACCESS

for X68000

V70アクセラレータがどうして速いのか
その秘密をこっそり教えて上げましょう。
それは賢い小人さんがたくさん住んでいるからなのです。
この小人さん達はお絵描きが大好きです。
グラフィック演算はまかせなさい！
レイトレーシングだってあっという間にほらっできあがり。
V70とX68000との面倒なやりとりもみーんなやってくれるので
材料さえ渡しておけばあら簡単！
靴屋のおじいさんのように「眠っている間に」ではなく、
あなたが「お茶を飲んでいる間に」出来上がっているというわけです。
あなたになら小人さん達の頑張っている姿が見えるかも知れませんね。

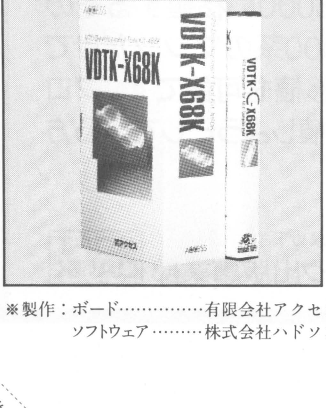


誰も知らなかった



V70のヒ・ミ・ツ

V70 アクセラレータ　VDTK-X68K



※製作：ボード……………有限会社アクセス
　　　ソフトウェア………株式会社ハドソン

VDTK-X68Kの仕様

- ●V70 CPU(μPD70632)
  20MHz 32ビットマイクロプロセッサ
- ●V70AFPP(μPD72691)
  フローティング・ポイント・プロセッサ
- ●メインメモリ(DRAM)2Mバイト
  同一ページ内のアクセスはNo Wait
- ●共有メモリ(SRAM)128Kバイト
  X68000との通信用
- ●併行動作　X68000とV70は、併行に動作することが可能。
  データの受け渡し処理のために双方向ハンドシェークI/Oポートを搭載。

同梱ソフトウェア

- ●アセンブラ
- ●リンカ
- ●ソースコードデバッガ
- ●システムモニタ
- ●フロートエミュレータ
- ●コマンドシェル

オプションソフトウェア

- ●Cコンパイラ
  (VDTK-C-X68K)

価格

●ボードパッケージ　(XVI対応)
VDTK-X68K……………………**¥248,000**

●オプションソフト (Cコンパイラ)
VDTK-C-X68K……………………**¥68,000**

購入方法

上記商品は当面の間、通信販売のみとさせて頂きます。
購入ご希望の方は、住所、(社名、所属)氏名、電話番号をお知らせ下さい。注文書をお送りいたします。

●社員募集のお知らせ：アクセスでは技術者を募集しております。担当(田村)までご連絡下さい。

有限会社アクセス　〒101 東京都千代田区神田神保町1-64
神保町協和ビル7F
☎03(3233)0200(代)　FAX.03(3291)7019

資料請求券
oh!X
10月号

パソコン/ワープロ通信ネットワークサービス

J&P HOT LINE

タイムトラベルシリーズ

第7回

《もし、この時代にパソコン通信があったなら》

永遠なもの、それは少年の冒険心。

トム・ソーヤ

蒸気の外輪船が浮かぶ川のある小さな町、セント・ピーターズバーグ。ここで、われらがトム・ソーヤは活躍します。聖書の暗記は大の苦手。優等生の弟シッドに、いたずらをいいつけられて、ポーリーおばさんにしかられます。でも、このいたずらが、トム・ソーヤの元気のもと。大の仲良し、浮浪児のハックル・ベリーフィンと、家出をして海賊ごっこ、宝探しに殺人事件の証言と、ひらめきと行動力で次々に事件を解決し、いたずら小僧から町の英雄へと成長していきます。もちろんガールフレンドのベッキーと洞窟をさまよう大冒険も。永遠の少年の理想像。それがトム・ソーヤかもしれません。

もし、この時代にパソコン通信があったなら。

さて、もしこの時代にパソコン通信があったなら。人口100人前後のセント・ピーターズバーグでも、あれだけの大冒険をしでかしたトムのこと。人口何万人というパソコン通信の世界では、もっと画期的で胸のすくような大冒険をしでかしていたにちがいありません。悪者たちのたくらみを先取りして、解決した手腕は、全世界規模のひろがりと、トムの時代では考えられなかったようなスピードで発揮されていたはずです。でも、トムの大好きだった、ほほえましい「いたずら」も、うんと増えてしまったかもしれませんけどね。

パソコン通信なら、こんな楽しさ。

パソコン通信では、あらゆるテーマの情報が、データベースとして、世間話として飛びかっています。冒険心と行動力さえあれば、ネットワークは無限の「遊び場」。少年のようなワクワクした気持ちをとりもどしてくれる、それがパソコン通信なのかもしれません。

きっと、出会える。キットで、会える。

買ったその日から2週間無料でアクセスできます。

J&P HOTLINEへのご入会はスタータキットで。お求めは、下記のお店でどうぞ。または現金書留にて、¥3,000+¥90（消費税3%）=¥3,090を、事務局までお送り下さい。すぐにスタータキットをお送りします。

お問い合わせは――
〒556 大阪市浪速区日本橋西1-6-5 上新電機株式会社
J&P HOTLINE事務局宛 TEL(06) 632-2512

スタータキットのお求めはJ&P各店でどうぞ。

渋谷店	東京都渋谷区道玄坂2－28－4	☎(03)3496-4141
町田店	東京都町田市森野1－39－16	☎(0427)23-1313
八王子店	東京都八王子市旭町1-1八王子そごう7F	☎(0426)26-4141
立川店	東京都立川市幸町4－39－1	☎(0425)36-4141
三鷹店	東京都三鷹市野崎1－20－17	☎(0422)31-6251
横浜店	横浜市西区北幸2-9-5横浜HSビル1F	☎(045)313-6711
焼津インター店	静岡県焼津市越後島385	☎(054)626-3311
富山店	富山市掛尾町300	☎(0764)22-5033
金沢店	金沢市入江2－63	☎(0762)91-1130
寺地店	金沢市寺地2－3	☎(0762)47-2524
大須店	名古屋市中区大須4－2－48	☎(052)262-1141
テクノランド	大阪市浪速区日本橋5－6－7	☎(06) 634-1211
メディアランド	大阪市浪速区日本橋5－8－26	☎(06) 634-1511
コスモランド	大阪市浪速区難波中2－1－17	☎(06) 634-3111
U. S. LAND	大阪市浪速区日本橋4－9－15	☎(06) 634-1411
ビジネスランド	大阪市北区梅田1-1-3大阪駅前第3ビルB2	☎(06) 348-1881
高槻店	高槻市高槻町11－16	☎(0726)85-1212
くずは店	枚方市楠葉花園町15－2	☎(0720)56-8181
千里中央店	豊中市新千里東町1-3 SENCHU PAL 2番街4F	☎(06) 834-4141
摂津富田店	高槻市大畑町24－10	☎(0726)93-7521
寝屋川店	寝屋川市緑町4－20	☎(0720)34-1166
枚方バイパス店	枚方市田口3－41－7	☎(0720)48-1211
藤井寺店	藤井寺市岡2－1－33	☎(0729)38-2111
岸和田店	岸和田市土生町2451－3	☎(0724)37-1021
さんのみや1ばん館	神戸市中央区八幡通3－2－16	☎(078)231-2111
西宮店	西宮市河原町5－11	☎(0798)71-1171
伊丹店	伊丹市昆陽池1－63	☎(0727)77-5101
姫路店	姫路市東延末1-1住友生命姫路南ビル1F	☎(0792)22-1221
京都寺町店	京都市下京区寺町通仏光寺下ル恵比須之町549	☎(075)341-4411
京都近鉄店	京都市下京区烏丸通七条下ル東塩小路町702	☎(075)341-5769
和歌山店	和歌山市元寺町4－4	☎(0734)28-1441
和歌山南店	和歌山市中島368	☎(0734)25-1414
学園前店	奈良市学園北1－8－10	☎(0742)49-1411
奈良1ばん館	奈良市三条町478－1	☎(0742)27-1111
新大宮店	奈良市法華寺町83－5	☎(0742)35-2611
郡山インター店	大和郡山市横田693－1	☎(07435)9-2221
熊本店	熊本市手取本町4－12	☎(096)359-7800

SHARP

目の付けどころが、
シャープでしょ。

いわば“感性”専用。

ことマインドに関しては
「汎用」という概念は存在しないも同じです。
「実用的である」のと、これなら「使える」というのも違います。
X68000が、普通のパソコンとは違うといわれる所以もここにあります。
いわゆる実用性を重視したビジネスパソコンとは
創造力で一線を画しています。
何に使うのか、何がしたいのか、
パソコン選びのポイントは目的にあったマシンを探すこと。
普通のパソコンに合わせるのでは
あなたのせっかくの創造力も発揮されません。
X68000は、使う人のクリエイティブマインドを咲かせる
“感性”専用パソコンです。

X68000
PERSONAL WORKSTATION・XVI
Compact

本体＋キーボード＋マウス
2HD3.5インチFDDタイプ CZ-674C-H（グレー） 標準価格298,000円（税別）
14型カラーディスプレイ（ドットピッチ0.28mm）
CZ-608D-H（グレー） 標準価格94,800円（税別）

●5.25インチ増設用フロッピーディスクドライブ CZ-6FD5 標準価格99,800円・税別〔接続ケーブル同梱〕
●ディスプレイテレビ/CZ-6TU用RGBケーブル CZ-6CR1 標準価格4,500円・税別
●ディスプレイテレビ/CZ-6TU用テレビコントロールケーブル CZ-6CT1 標準価格5,500円・税別
●SCSI変換ケーブル CZ-6CS1 標準価格12,000円・税別

（カラー液晶ディスプレイとの組み合わせ例）
10.4型TFTカラー液晶ディスプレイ
LC-10C1-H（グレー）標準価格598,000円（税別）
接続ケーブル AN-1515X 標準価格4,200円（税別）
※カラー液晶ディスプレイを接続してご使用の場合、SX-WINDOW上のアプリケーション利用に限定されます。

●お問い合わせは…
シャープ株式会社 電子機器事業本部システム機器営業部〒545大阪市阿倍野区長池町22番22号☎(06)621-1221（大代表） 電子機器事業本部AVCシステム事業推進室〒162東京都新宿区市谷八幡町8番地☎(03)3260-1161（大代表）

Oh!X
10月号 1992年10月1日発行（毎月1回1日発行）通巻126号 昭和58年11月2日第三種郵便物認可 ソフトバンク株式会社発行 Printed in Japan 定価600円（本体583円）

T1002179100603 雑誌 02179-10